The Ensemble Kalman filter and related filters

Inge Myrseth & Henning Omre

Norwegian University of Science and Technology

1.1 Introduction

Temporal phenomena are abundant in nature and in man-created activity. Traditionally, mathematicians have modeled these phenomena by differential equations while statisticians have relied on empirically based time series models. It is a natural challenge to combine these two modeling approaches, and hidden Markov models have proven efficient in doing so. The inherent local characteristics of differential equations justifies the Markov assumption while the empirical data is linked to the variables of interest through likelihood functions. Evaluation of the hidden Markov model can be done by Bayesian inversion.

R. E. Kalmans' celebrated paper (Kalman 1960) was based on this line of thought. Under very specific assumptions about linearity and Gaussianity exact analytical solutions can be determined for the Bayesian inversion. Whenever deviations from these assumptions occur however, one has to rely on approximations. This gives room for a large variety of approaches including linearizations and simulation based inference.

The current paper focuses on simulation based inference of hidden Markov models and on ensemble Kalman filters in particular. The ensemble Kalman filter was introduced by Evensen in the papers Evensen (1994) and Burgers et al. (1998). The

Computational Methods for Large-Scale Inverse Problems and Quantification of Uncertainty. Edited by L. Biegler, G. Biros, O. Ghattas, M. Heinkenschloss, D. Keyes, B. Mallick, L. Tenorio, B. van Bloemen Waanders, K. Willcox

filter relies on simulation based inference and utilizes a linearization in the data conditioning. These approximations make the ensemble Kalman filter computationally efficient and well suited for high-dimensional hidden Markov models. Hence this filter has found widespread use in evaluation of spatio-temporal phenomena like ocean modeling, weather forecasting and petroleum reservoir evaluation, see Bertino et al. (2002), Houtekamer et al. (2005), Nævdal et al. (2005), and references therein.

The ensemble Kalman filter and its characteristics are the major theme of this paper, but also related simulation based filters like the randomized maximum likelihood filter and the particle filter are briefly defined. The properties of the various filters are demonstrated on a small example.

1.2 Model assumptions

Consider an unknown, multivariate time series $[\mathbf{x}_0, \mathbf{x}_1, ..., \mathbf{x}_T, \mathbf{x}_{T+1}]$ with $\mathbf{x}_t \in \mathbb{R}^{p_x}$; $t = 0, ..., T+1$ containing the primary variable of interest and $\mathbf{x}_T$ being the current state. Assume that an associated time series of observations $[\mathbf{d}_0, ..., \mathbf{d}_T]$ with $\mathbf{d}_t \in \mathbb{R}^{p_d}$; $t = 0, ..., T$, is available. The primary objective of the study is to assess the forecasting problem, namely evaluate $\mathbf{x}_{T+1}$ given $[\mathbf{d}_0, ..., \mathbf{d}_T]$.

Define a prior stochastic model for $[\mathbf{x}_0, ..., \mathbf{x}_{T+1}]$ by assuming Markov properties:

$$\begin{aligned}
[\mathbf{x}_0, ..., \mathbf{x}_{T+1}] &\sim f(\mathbf{x}_0, ..., \mathbf{x}_{T+1}) \\
&= f(\mathbf{x}_0) \prod_{t=0}^{T} f(\mathbf{x}_{t+1} | \mathbf{x}_0, ..., \mathbf{x}_t) \\
&= f(\mathbf{x}_0) \prod_{t=0}^{T} f(\mathbf{x}_{t+1} | \mathbf{x}_t),
\end{aligned}$$

with $\mathbf{x} \sim f(\mathbf{x})$ reading "the random variable $\mathbf{x}$ is distributed according to the probability density function (pdf) $f(\mathbf{x})$". Let $f(\mathbf{x}_0)$ be a known pdf for the initial state, and $f(\mathbf{x}_{t+1}|\mathbf{x}_t)$; $t = 0, ..., T$ be known forward pdfs. Hence the prior model for the time series of interest is Markovian with each state given the past dependent on the previous state only.

Define the likelihood model for $[\mathbf{d}_0, ..., \mathbf{d}_T]$ given $[\mathbf{x}_0, ..., \mathbf{x}_{T+1}]$ by assuming conditional independence and single state dependence:

$$\begin{aligned}
[\mathbf{d}_0, ..., \mathbf{d}_T | \mathbf{x}_0, ..., \mathbf{x}_{T+1}] &\sim f(\mathbf{d}_0, ..., \mathbf{d}_T | \mathbf{x}_0, ..., \mathbf{x}_{T+1}) \\
&= \prod_{t=0}^{T} f(\mathbf{d}_t | \mathbf{x}_0, ..., \mathbf{x}_{t+1}) \\
&= \prod_{t=0}^{T} f(\mathbf{d}_t | \mathbf{x}_t)
\end{aligned}$$

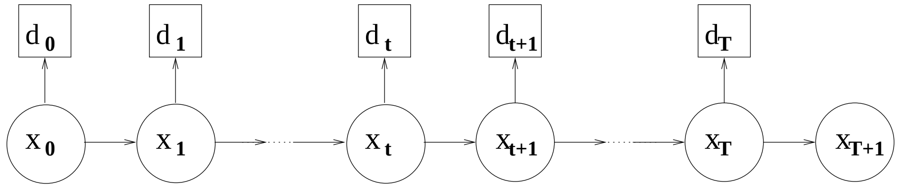


Figure 1.1 Hidden Markov process

where $f(\mathbf{d}_t|\mathbf{x}_t)\,;\,t = 0, ..., T$ are known likelihood functions. Hence, the likelihood model entails that the observations at time t is a function of state $\mathbf{x}_t$ only and is independent of the other observations when $\mathbf{x}_t$ is given.

These prior and likelihood assumptions define a hidden Markov process as depicted by the diagram in Figure 1.1. The arrows in the graph represent causal, stochastic dependencies between nodes. The resulting posterior stochastic model is defined by Bayesian inversion:

$$\begin{aligned}[\mathbf{x}_0, ..., \mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T] \sim\, & f(\mathbf{x}_0, ..., \mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T)\\ &= \text{const} \times f(\mathbf{d}_0, ..., \mathbf{d}_T|\mathbf{x}_0, ..., \mathbf{x}_{T+1}) \times f(\mathbf{x}_0, ..., \mathbf{x}_{T+1})\\ &= \text{const} \times f(\mathbf{x}_0) f(\mathbf{d}_0|\mathbf{x}_0) \left[\prod_{t=0}^{T-1} f(\mathbf{d}_{t+1}|\mathbf{x}_{t+1}) f(\mathbf{x}_{t+1}|\mathbf{x}_t)\right]\\ &\quad \times f(\mathbf{x}_{T+1}|\mathbf{x}_T),\end{aligned}$$

with 'const' being a normalizing constant that is usually hard to assess. Hence the full posterior model is not easily available.

The forecasting problem is the major objective of this study. The forecasting pdf is:

$$\begin{aligned}[\mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T] &\sim f(\mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T)\\ &= \int ... \int f(\mathbf{x}_0, ..., \mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T) \mathrm{d}\mathbf{x}_0 ... \mathrm{d}\mathbf{x}_T.\end{aligned}$$

This forecasting pdf is computable by a recursive algorithm. In order to simplify notation, introduce:

$$\begin{aligned}\mathbf{x}_t^u &= [\mathbf{x}_t|\mathbf{d}_0, ..., \mathbf{d}_{t-1}]\\ \mathbf{x}_t^c &= [\mathbf{x}_t|\mathbf{d}_0, ..., \mathbf{d}_t],\end{aligned}$$

where indices u and c indicate unconditioned and conditioned on the observation at the current time, respectively. The recursive algorithm can be justified by the graph in Figure 1.1, and it is defined as:

Algorithm 1: Recursive forecasting

- Initiate:

$$\mathbf{x}_0^u \sim f(\mathbf{x}_0^u) = f(\mathbf{x}_0)$$

- Iterate $t = 0, ..., T$

Conditioning:

$$\mathbf{x}_t^c \sim f(\mathbf{x}_t^c) = f(\mathbf{x}_t^u|\mathbf{d}_t) = \text{const} \times f(\mathbf{d}_t|\mathbf{x}_t^u)f(\mathbf{x}_t^u)$$

Forwarding:

$$\mathbf{x}_{t+1}^u \sim f(\mathbf{x}_{t+1}^u) = \int f(\mathbf{x}_{t+1}^u|\mathbf{x}_t^c)f(\mathbf{x}_t^c)\mathrm{d}\mathbf{x}_t^c$$

- end iterate

$$\mathbf{x}_{T+1}^u = [\mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T] \sim f(\mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T) = f(\mathbf{x}_{T+1}^u)$$

Hence the forecast pdf $f(\mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T)$, which is the objective of the study, is obtained at time step $t = T + 1$. The recursion relies on a conditioning operation and a forwarding operation at each step. Note further that this recursive algorithm makes sequential conditioning on future observations possible.

Algorithm 1 relies on assumptions about independence made for the hidden Markov process, but beyond this no specific distributional assumptions are made. This entails that the prior model can be written as:

$$\begin{aligned} \mathbf{x}_0 &\sim f(\mathbf{x}_0) \\ [\mathbf{x}_{t+1}|\mathbf{x}_t] &= \omega_t(\mathbf{x}_t, \boldsymbol{\epsilon}_t^x) \sim f(\mathbf{x}_{t+1}|\mathbf{x}_t) \end{aligned} \tag{1.1}$$

where $\omega_t(.,.)$ is a known function $\mathbb{R}^{2p_x} \to \mathbb{R}^{p_x}$ and $\boldsymbol{\epsilon}_t^x$ is a random variable from the normalized p_x-dimensional multivariate Gaussian distribution $N_{p_x}(\mathbf{0}, \mathbf{I}_{p_x})$ where $\mathbf{I}_{p_x}$ is a unit diagonal covariance matrix. This construction can generate a realization from an arbitrary $f(\mathbf{x}_{t+1}|\mathbf{x}_t)$. The likelihood model may be constructed in a similar manner:

$$[\mathbf{d}_t|\mathbf{x}_t] = \nu_t(\mathbf{x}_t, \boldsymbol{\epsilon}_t^d) \sim f(\mathbf{d}_t|\mathbf{x}_t), \tag{1.2}$$

where $\nu_t(.,.)$ is a known function $\mathbb{R}^{p_x+p_d} \to \mathbb{R}^{p_d}$ and $\boldsymbol{\epsilon}_t^d$ is a normalized p_d- dimensional Gaussian random variable from $N_{p_d}(\mathbf{0}, \mathbf{I}_{p_d})$.

The associated Gauss-linear model is defined as:

$$\begin{aligned} \mathbf{x}_0 &\sim f(\mathbf{x}_0) = N_{p_x}(\boldsymbol{\mu}_0^x, \boldsymbol{\Sigma}_0^x) \\ [\mathbf{x}_{t+1}|\mathbf{x}_t] &= \mathbf{A}_t\mathbf{x}_t + \boldsymbol{\epsilon}_t^x \sim f(\mathbf{x}_{t+1}|\mathbf{x}_t) = N_{p_x}(\mathbf{A}_t\mathbf{x}_t, \boldsymbol{\Sigma}_t^x) \\ [\mathbf{d}_t|\mathbf{x}_t] &= \mathbf{H}_t\mathbf{x}_t + \boldsymbol{\epsilon}_t^d \sim f(\mathbf{d}_t|\mathbf{x}_t) = N_{p_d}(\mathbf{H}_t\mathbf{x}_t, \boldsymbol{\Sigma}_t^d), \end{aligned} \tag{1.3}$$

where $\mathbf{A}_t$ and $\mathbf{H}_t$ are known matrices of proper dimensions and the error term is additive and Gaussian independent of $\mathbf{x}_t$.

Under the Gauss-linear assumptions the recursive Algorithm 1 is analytically tractable, and this analytical solution corresponds to the traditional Kalman filter (KF) which will be presented below. For other model assumptions the hidden Markov process must be assessed by some sort of approximation for which the recursive Algorithm 1 is well suited. In the current paper we will focus on simulation based approximate solutions to the forecast problem.

Two Test Examples

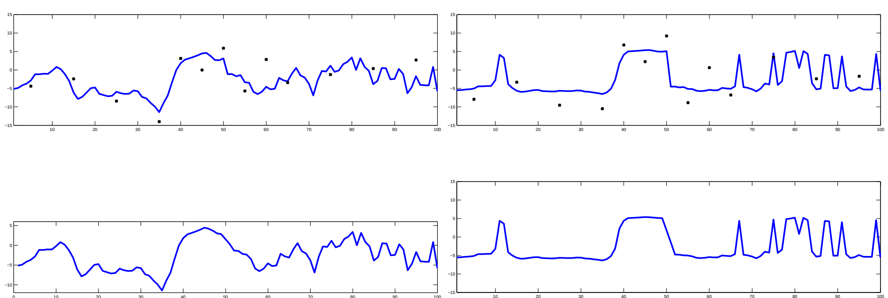
In order to illustrate the basic characteristics of the various filters, we define two simple examples that will be used throughout this paper. The first example is based on a Gauss-linear model, termed linear case, and it is subject to both analytical treatment and simulation based inference. The other case is based on a model with nonlinearities, termed nonlinear case, and it can only be evaluated by simulation based inference.

The variable of interest is $[\mathbf{x}_0, ..., \mathbf{x}_{11}]$, where $\mathbf{x}_t \in \mathbb{R}^{100}$; hence $\mathbf{x}_t$ is a 100-dimensional time series. Observations are available at $[\mathbf{d}_0, ..., \mathbf{d}_{10}]$. The current time is $T = 10$ and the objective is the forecast $[\mathbf{x}_{11}|\mathbf{d}_0, ..., \mathbf{d}_{10}]$. In Figure 1.2 the reference realizations of $[\mathbf{x}_{10}, \mathbf{d}_{10}]$ and $\mathbf{x}_{11}$ for both the linear and nonlinear cases are presented. The dimensions of $\mathbf{x}_t$ are denoted by nodes.

The linear case is defined as follows:

$$
\begin{aligned}
f(\mathbf{x}_0) &\sim N_{100}(\mathbf{0}, \mathbf{\Sigma}_0^x) \\
[\mathbf{x}_{t+1}|\mathbf{x}_t] &= \mathbf{A}_t \mathbf{x}_t \\
[\mathbf{d}_t|\mathbf{x}_t] &= \mathbf{H}_t \mathbf{x}_t + \boldsymbol{\epsilon}_t^d \sim f(\mathbf{d}_t|\mathbf{x}_t) = N_{13}(\mathbf{H}_t\mathbf{x}_t, \mathbf{\Sigma}_t^d),
\end{aligned}
$$

where the initial covariance matrix $\mathbf{\Sigma}_0^x$ is constructed from a covariance function $c(\Delta) = 20\exp(-3\Delta/20)$ with Δ being distance between nodes in $\mathbf{x}_0$. The forward



Linear case

Nonlinear case

Figure 1.2 Reference realizations: $(\mathbf{x}_{10}, \mathbf{d}_{10})$ in the upper display and $\mathbf{x}_{11}$ in the lower display.

model defined by $\mathbf{A}_t$ is a uniform smoother of width 10 nodes that moves in steps of 5 from left to right for each time step. Consequently, the left part of $\mathbf{x}_{10}$ is smoother than the right part. The likelihood model defined by $(\mathbf{H}_t, \boldsymbol{\Sigma}_t^d)$ returns 13 observations with independent errors of variance 10 at each time step, see Figure 1.2.

The nonlinear case is defined as follows:

$$\begin{aligned}
f(\mathbf{x}_0) &\sim N_{100}(\mathbf{0}, \boldsymbol{\Sigma}_0^x) \\
[\mathbf{x}_{t+1}|\mathbf{x}_t] &= c\mathbf{A}_t(\mathbf{x}_t + \arctan(\mathbf{x}_t)) \\
[\mathbf{d}_t|\mathbf{x}_t] &= \mathbf{H}_t(\mathbf{x}_t + \arctan(\mathbf{x}_t)) + \boldsymbol{\epsilon}_t^d \sim f(\mathbf{d}_t|\mathbf{x}_t) \\
&= N_{13}(\mathbf{H}_t(\mathbf{x}_t + \arctan(\mathbf{x}_t)), \boldsymbol{\Sigma}_t^d),
\end{aligned}$$

where the model parameters are defined mostly as for the linear case, except for c being a scaling factor to align the variances in the linear and nonlinear case, and for $\arctan(.)$ being a functional that acts element-wise on $\mathbf{x}_t$. As Figure 1.2 shows, the nonlinear example appears more box-like than the linear one.

1.3 The Traditional Kalman Filter (KF)

The KF was introduced in Kalman (1960). The hidden Markov process was made analytically tractable by the recursive Algorithm 1 and the Gauss-linear model assumptions in Expression (1.3) with fully known parameter values.

The recursive algorithm will under these Gauss-linear assumptions reproduce Gaussianity from one step to the next, and the algorithm appears as:

Algorithm 2: Kalman Filter

- Initiate:

$$\begin{aligned}
&\mathbf{x}_0^u \sim f(\mathbf{x}_0^u) = N_{p_x}(\boldsymbol{\mu}_0^u, \boldsymbol{\Sigma}_0^u) \\
&\boldsymbol{\mu}_0^u = \boldsymbol{\mu}_0^x \\
&\boldsymbol{\Sigma}_0^u = \boldsymbol{\Sigma}_0^x
\end{aligned}$$

- Iterate $t = 0, ..., T$

Conditioning:

$$\begin{aligned}
&\mathbf{x}_t^c \sim f(\mathbf{x}_t^c) = N_{p_x}(\boldsymbol{\mu}_t^c, \boldsymbol{\Sigma}_t^c) \\
&\boldsymbol{\mu}_t^c = \boldsymbol{\mu}_t^u + \boldsymbol{\Sigma}_t^u \mathbf{H}_t' [\mathbf{H}_t' \boldsymbol{\Sigma}_t^u \mathbf{H}_t' + \boldsymbol{\Sigma}_t^d]^{-1} (\mathbf{d}_t - \mathbf{H}_t \boldsymbol{\mu}_t^u) \\
&\boldsymbol{\Sigma}_t^c = \boldsymbol{\Sigma}_t^u - \boldsymbol{\Sigma}_t^u \mathbf{H}_t' [\mathbf{H}_t \boldsymbol{\Sigma}_t^u \mathbf{H}_t' + \boldsymbol{\Sigma}_t^d]^{-1} \mathbf{H}_t \boldsymbol{\Sigma}_t^u
\end{aligned}$$

Forwarding:

$$\begin{aligned}
&\mathbf{x}_{t+1}^u \sim f(\mathbf{x}_{t+1}^u) = N_{p_x}(\boldsymbol{\mu}_{t+1}^u, \boldsymbol{\Sigma}_{t+1}^u) \\
&\boldsymbol{\mu}_{t+1}^u = \mathbf{A}_t \boldsymbol{\mu}_t^c
\end{aligned}$$

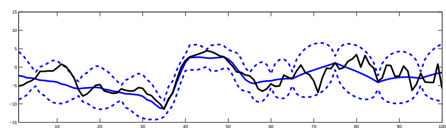
$$\boldsymbol{\Sigma}_{t+1}^{u} = \mathbf{A}_t \boldsymbol{\Sigma}_t^{c} \mathbf{A}_t' + \boldsymbol{\Sigma}_t^{x}$$

- end iterate
- $f(\mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T) = N_{p_x}(\boldsymbol{\mu}_{T+1}^{u}, \boldsymbol{\Sigma}_{T+1}^{u})$

Consequently, all relevant pdfs are analytically tractable and no approximations of the forecast problem is needed. Whenever deviations from these Gauss-linear assumptions arise however, the analytical tractability is lost. Approximate solutions can of course be made, and different approximations have defined a large family of Kalman Filter variations such as the extended Kalman filter (Jazwinski 1970), the unscented Kalman Filter (Julier and Uhlmann 1997) and others. These approximations follow the KF tradition by focusing on the two first moments of the pdf as they develop through time. Since traditional Kalman filters are not the focus of this presentation, we leave the reader with these references and the applications to the test problems below.

Performance on the Test Examples

The traditional KF can only be applied to the linear case, and the solution is shown in Figure 1.3 together with the $\mathbf{x}_{11}$ reference realization. The KF prediction represented by $\mathrm{E}\{\mathbf{x}_{11}|\mathbf{d}_0, ..., \mathbf{d}_{10}\}$ and associated 0.95 prediction intervals are displayed. This KF solution is the exact solution and can be computed analytically.



Linear case

Figure 1.3 Results of the KF algorithm on the linear case: reference realization $\mathbf{x}_{11}$ (black), prediction (blue) and .95-prediction intervals (hatched blue).

1.4 The Ensemble Kalman Filter (EnKF)

The EnKF provides an approximate solution to the forecast problem both under Gauss-linear model assumptions and whenever deviations from these assumptions occur. The basic idea of the EnKF is to create a set of realizations, so called ensemble, from the initial model. These realizations are adjusted according to the likelihood model when an observation occurs and the adjusted realizations are then taken through the forward model to the next observation time. At time $t = T + 1$ a set of approximately independent realizations are available for empirical assessment of

$f(\mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T)$. Hence, characteristics beyond the two first moments can be captured. Note also that the actual adjustment of realizations as observations appear makes it meaningful to visually inspect the ensemble members to seek for common features. Basic references for EnKF are Evensen (1994), Burgers et al. (1998), Evensen (2007) and references therein.

A time series of ensembles is defined by:

$$\mathbf{e}_t \,:\, \{(\mathbf{x}^u_t, \mathbf{d}_t)^{(i)}; i = 1, ..., n_e\}\,;\, t = 0, ..., T + 1,$$

where $\mathbf{x}^{u(i)}_t = [\mathbf{x}_t|\mathbf{d}_0, ..., \mathbf{d}_{t-1}]^{(i)}$ are approximate realizations from $f(\mathbf{x}_t|\mathbf{d}_0, ..., \mathbf{d}_{t-1})$ and $\mathbf{d}^{(i)}_t$ are associated realizations of the observation available at time t, i.e., $\mathbf{d}_t$. Note that at any step t - with t omitted in the notation - one has the expectation vector and covariance matrix:

$$\boldsymbol{\mu}_{xd} = \begin{bmatrix} \boldsymbol{\mu}_x \\ \boldsymbol{\mu}_d \end{bmatrix}$$
$$\boldsymbol{\Sigma}_{xd} = \begin{bmatrix} \boldsymbol{\Sigma}_x & \boldsymbol{\Gamma}_{x,d} \\ \boldsymbol{\Gamma}_{d,x} & \boldsymbol{\Sigma}_d \end{bmatrix}.$$

The EnKF algorithm in its general form is based on the recursive forecasting algorithm, Algorithm 1, and appears as:

Algorithm 3: Ensemble Kalman filter (general)

- Initiate

$$n_e = \text{ no. of ensemble members}$$
$$\mathbf{x}^{u(i)}_0\,;\, i = 1, ..., n_e \text{ iid } f(\mathbf{x}_0)$$
$$\boldsymbol{\epsilon}^{d(i)}_0 \sim N_{p_d}(\mathbf{0}, \mathbf{I}_{p_d})\,;\, i = 1, ..., n_e$$
$$\mathbf{d}^{(i)}_0 = \nu_t(\mathbf{x}^{u(i)}_0, \boldsymbol{\epsilon}^{d(i)}_0)\,;\, i = 1, ..., n_e$$
$$\mathbf{e}_0 \,:\, \{(\mathbf{x}^u_0, \mathbf{d}_0)^{(i)}\,;\, i = 1, ..., n_e\}$$

- Iterate $t = 0, ..., T$

Conditioning:

Estimate $\boldsymbol{\Sigma}_{xd}$ from $\mathbf{e}_t \rightarrow \widehat{\boldsymbol{\Sigma}}_{xd}$

$$\mathbf{x}^{c(i)}_t = \mathbf{x}^{u(i)}_t + \widehat{\boldsymbol{\Gamma}}_{x,d}\widehat{\boldsymbol{\Sigma}}^{-1}_d(\mathbf{d}_t - \mathbf{d}^{(i)}_t)\,;\, i = 1, ..., n_e$$

Forwarding:

$$\boldsymbol{\epsilon}^{x(i)}_t \sim N_{p_x}(\mathbf{0}, \mathbf{I}_{p_x})\,;\, i = 1, ..., n_e$$
$$\mathbf{x}^{u(i)}_{t+1} = \omega_t(\mathbf{x}^{c(i)}_t, \boldsymbol{\epsilon}^{x(i)}_t)\,;\, i = 1, ..., n_e$$
$$\boldsymbol{\epsilon}^{d(i)}_{t+1} \sim N_{p_d}(\mathbf{0}, \mathbf{I}_{p_d})\,;\, i = 1, ..., n_e$$

$$\mathbf{d}_{t+1}^{(i)} = \nu_{t+1}(\mathbf{x}_{t+1}^{u(i)}, \boldsymbol{\epsilon}_{t+1}^{d(i)})\,;\, i = 1, ..., n_e$$
$$\mathbf{e}_{t+1} \,:\, \{(\mathbf{x}_{t+1}^{u}, \mathbf{d}_{t+1})^{(i)}\,;\, i = 1, ..., n_e\}$$

- end iterate
- Assess

$$f(\mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T) \text{ from } \mathbf{e}_{T+1}$$

The resulting ensemble $\mathbf{e}_{T+1}$ contains approximately independent realizations of

$$\mathbf{x}_{T+1}^{u} = [\mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T]$$

from $f(\mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T)$ which can be used to assess the forecast pdf. If the forecast expectation vector and covariance matrix are of interest regular estimators can be used.

$$\widehat{\boldsymbol{\mu}}_{T+1} = \widehat{\mathrm{E}}\{\mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T\} = \frac{1}{n_e}\sum_{i=1}^{n_e} \mathbf{x}_{T+1}^{u(i)} \tag{1.4}$$

$$\widehat{\boldsymbol{\Sigma}}_{T+1} = \widehat{\mathrm{Var}}\{\mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T\} = \frac{1}{n_e - 1}\sum_{i=1}^{n_e} (\mathbf{x}_{T+1}^{u(i)} - \widehat{\boldsymbol{\mu}}_{T+1})(\mathbf{x}_{T+1}^{u(i)} - \widehat{\boldsymbol{\mu}}_{T+1})'.$$

The EnKF algorithm, Algorithm 3, is recursive and each recursion consists of a conditioning operation and a forwarding operation. The conditioning expression is linear with weights estimated from the ensemble. The forwarding operation is defined by the forward pdf.

There are two implicit approximations in the EnKF:

1. Discretization of the sample space of $\mathbf{x}_t$. The initial ensemble of iid realizations is assumed to represent $f(\mathbf{x}_0)$. For high-dimensional problems a large number of ensemble members may be required to do so reliably.
2. The data conditioning expression is linearized. Moreover, the weights in the linearization are estimated from the ensemble. Note, however, that each ensemble member is conditioned individually and hence the linearization only applies to the conditioning not to the forward model. For highly non-Gaussian prior models and/or strongly nonlinear likelihood models this approximation may provide unreliable results.

Under these approximations, however, all types of models for the hidden Markov process can be evaluated. The EnKF is a consistent forecast procedure under a Gauss-linear model in the sense that the exact solution is obtained as $n_e \to \infty$ (see Appendix A).

A graphical description of the EnKF is shown in Figure 1.4. At iteration t, the ensemble $\mathbf{e}_t$ with $n_e = 5$ is shown in display (a) together with the observed value $\mathbf{d}_t$. The regression slope of $\mathbf{d}_t$ on $\mathbf{x}_t$ is estimated by $\widehat{\boldsymbol{\Gamma}}_{x,d}\widehat{\boldsymbol{\Sigma}}_d^{-1}$ and this corresponds

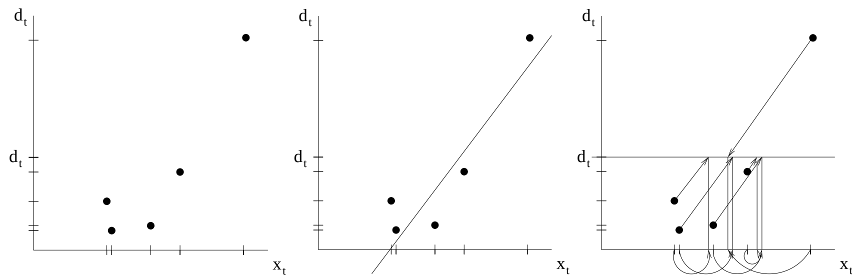


Figure 1.4 Graphical description of one step in the EnKF algorithm; ensemble (a), linearization (b) and ensemble conditioning (c). See explanation in text.

to the regression slope in display (b). The EnKF-update is made on each ensemble member by using the estimated regression slope as presented in display (c). Note how the individual unconditioned members $\mathbf{x}_t^{u(i)}$ are moved to the conditioned one $\mathbf{x}_t^{c(i)}$. To establish the ensemble $\mathbf{e}_{t+1}$ at iteration $t+1$, each $\mathbf{x}_t^{c(i)}$ is taken through the forward model to obtain $\mathbf{x}_{t+1}^{u(i)}$ and thereafter each $\mathbf{x}_{t+1}^{u(i)}$ is taken through the likelihood model to obtain $\mathbf{d}_{t+1}^{(i)}$. This brings us back to display (a) for iteration $t+1$.

The EnKF is a consistent forecast procedure for Gauss-linear models, but frequently deviations from Gaussianity in the prior and likelihood model appear. Consequences of these deviations from Gaussianity are discussed in Section 1.4.1. Other problems arise in the EnKF which are caused by the use of an estimate of $\mathbf{\Sigma}_{xd}$ based on $\mathbf{e}_t$ instead of the true covariance matrix. These problems include rank deficiency and estimation uncertainty due to the limited size of the ensemble, i.e., small values of n_e. This issue will be discussed further in Section 1.4.2.

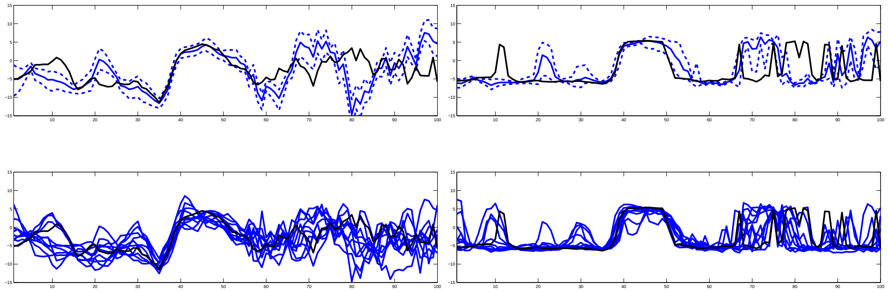
Performance on the Test Examples

The EnKF can be applied to both the linear and the nonlinear test cases. The solutions are presented in Figure 1.5 together with the $\mathbf{x}_{11}$ reference realization. The EnKF is based on the ensemble size $n_e = 100$, which is often used as rule of thumb.

The upper displays in Figure 1.5 show the predictions $\widehat{\mathrm{E}}\{\mathbf{x}_{11}|\mathbf{d}_0, ..., \mathbf{d}_{10}\}$ and associated 0.95 empirical prediction intervals. The former is obtained as the average of the 100 ensemble members, while the latter is defined to be the range between the third smallest and the third largest ensemble member at any node.

The lower displays show the predictions from ten different EnKF runs, each of these with $n_e = 100$.

For the linear case, the correct solution is the KF solution shown in Figure 1.3. The EnKF solution with $n_e = 100$ does reproduce the main features of the reference realization $\mathbf{x}_{11}$, but as seen in the upper display, it seems that the prediction



Linear case Nonlinear case

Figure 1.5 Results of the EnKF algorithm on the linear and nonlinear test cases: reference realization $\mathbf{x}_{11}$ (black), prediction (blue) and empirical .95-prediction intervals (hatched blue).

interval is underestimated. Some of the variability appears to be shifted to variability in between predictions based on different ensembles as seen in the lower display. Recall that the EnKF algorithm is consistent in the sense that the KF solution is reproduced in the limit as $n_e \to \infty$. For the nonlinear case, the picture is basically the same. The results indicate that it is important to have a reasonably high number of ensemble members.

1.4.1 Variable Characteristics

The linear conditioning in the EnKF is correct if at each step the likelihood model $f(\mathbf{d}_t|\mathbf{x}_t^u)$ is Gauss-linear and the prior model $f(\mathbf{x}_t^u)$ is Gaussian. Often we can control the observation procedure or reformulate the model and ensure approximate Gauss-linearity in the likelihood, hence the discussion will focus on the characteristics of $\mathbf{x}_t^u$.

The EnKF is expected to be reliable as long as $f(\mathbf{x}_t^u)$ is unimodal and reasonably symmetric since all elliptically symmetric pdfs have conditional models that are linear in the variables (Kelker 1970). If, however, the $f(\mathbf{x}_t^u)$ is multimodal, the picture may be very different. Usually, multimodality indicates that the variable of interest $\mathbf{x}_t^u$ is a nested variable with a discrete underlying variable defining different levels the variable may be on, and added to this level is a continuous variable representing locally smooth variations. The EnKF approach has no mechanism for jumping from one discrete outcome to the other; it fully relies on continuous adjustments through the linearized conditioning expression. Hence the EnKF solution, in cases with multimodal $f(\mathbf{x}_t^u)$, may deviate considerably from the correct solution $f(\mathbf{x}_t^u|\mathbf{d}_t)$. Several approaches to solving the multimodal problem have been suggested in the literature:

1. Mixtures of Gaussian pdfs may be used as model for $f(\mathbf{x}_t^u)$, and a nonlinear updating rule can be defined to obtain samples from $f(\mathbf{x}_t^u|\mathbf{d}_t)$ (Dovera

and Della Rossa 2007). The nonlinear updating rule will be computationally demanding and may not be suitable for high-dimensional problems.

2. Reparametrization of the discrete underlying variable into continuous variables, preferably with unimodal pdfs. In spatial problems where the discrete process may be related to domains in space, the boundary of these domains may be characterized by continuous variables. The conditioning can then be made on these boundary variables (Bertino et al. 2003; Liu and Oliver 2005a,b). The reparametrization is usually highly problem specific and the efficiency is hard to judge.

3. Basis-function expansion of the variables of interest $\mathbf{x}_t^u$. The associated coefficients are conditioned by the EnKF procedure (Jafarpour and McLaughlin 2007) . The choice of basis functions is crucial for the success of the approach and no clear guideline seems to be available so far.

1.4.2 Parameter Estimates

The EnKF algorithm depends crucially on the term $\mathbf{\Gamma}_{x,d}\mathbf{\Sigma}_d^{-1}$ which must be inferred from the ensemble $\mathbf{e}_t$ of size n_e. Usually the classical covariance matrix estimators are applied:

$$\widehat{\mathbf{\Gamma}}_{x,d} = \frac{1}{n_e-2}\sum_{i=1}^{n_e}(\mathbf{x}_t^{u(i)} - \widehat{\boldsymbol{\mu}}_x)(\mathbf{d}_t^{(i)} - \widehat{\boldsymbol{\mu}}_d)'$$

$$\widehat{\mathbf{\Sigma}}_d = \frac{1}{n_e-1}\sum_{i=1}^{n_e}(\mathbf{d}_t^{(i)} - \widehat{\boldsymbol{\mu}}_x)(\mathbf{d}_t^{(i)} - \widehat{\boldsymbol{\mu}}_x)',$$

with

$$\widehat{\boldsymbol{\mu}}_x = \frac{1}{n_e}\sum_{i=1}^{n_e}\mathbf{x}_t^{u(i)}$$

$$\widehat{\boldsymbol{\mu}}_d = \frac{1}{n_e}\sum_{i=1}^{n_e}\mathbf{d}_t^{(i)}.$$

These estimators are unbiased and consistent, i.e., $\widehat{\mathbf{\Gamma}}_{x,d} \to \mathbf{\Gamma}_{x,d}$ and $\widehat{\mathbf{\Sigma}}_d \to \mathbf{\Sigma}_d$ as $n_e \to \infty$, for all distributional models. For finite n_e, the estimates have minimum variance for Gauss-linear models. With only slight deviations from Gauss-linearity, however, the estimation variance may increase dramatically (Huber 1981). Moreover, the ranks of the estimated matrices $\widehat{\mathbf{\Gamma}}_{x,d}$ and $\widehat{\mathbf{\Sigma}}_d$ are n_e-2 and n_e-1, respectively. Hence the reliability of the estimates are highly dependent on the actual true model and the size of the ensemble. Two major problems may occur in using the EnKF:

1. Rank deficiency. Recall that $\mathbf{\Gamma}_{x,d}$ and $\mathbf{\Sigma}_d$ are, respectively, $p_x \times p_d$ and $p_d \times p_d$ matrices. Hence $n_e \geq p_d + 2$ is required to ensure full rank of the estimates $\widehat{\mathbf{\Gamma}}_{x,d}$ and $\widehat{\mathbf{\Sigma}}_d$. In simple time series applications this requirement is easy to meet. In spatio-temporal applications with computationally demanding forward models and many observations, as for example in petroleum reservoir engineering and meteorology, requiring $n_e \geq p_d + 2$ may be prohibited. With reduced rank, the expression $\widehat{\mathbf{\Gamma}}_{x,d}\widehat{\mathbf{\Sigma}}_d^{-1}$ in the EnKF updating will be ill-defined.

2. Estimation uncertainty. Estimators for second order moments like the covariance matrices $\mathbf{\Gamma}_{x,d}$ and $\mathbf{\Sigma}_d$ are notoriously unreliable due to extreme dependence on the tail-behavior of the underlying pdf. It is complicated to make wise bias/variance trade-offs without making definite distributional assumptions on which the estimator properties rely heavily. Lack of precision in $\widehat{\mathbf{\Gamma}}_{x,d}\widehat{\mathbf{\Sigma}}_d^{-1}$ in the EnKF updating may cause spurious values to appear in the conditioned $\mathbf{x}_t^c$. This effect may be worsened by nonlinear forward models. Moreover, since the same estimate $\widehat{\mathbf{\Gamma}}_{x,d}\widehat{\mathbf{\Sigma}}_d^{-1}$ is used in the updating of all ensemble members in $\mathbf{e}_t$, unwanted dependencies between the members may be introduced. Ideally, the estimation uncertainty in $\widehat{\mathbf{\Gamma}}_{x,d}$ and $\widehat{\mathbf{\Sigma}}_d$ should be accounted for in the final assessment of $f(\mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T)$, but this is not done in the current version of the EnKF.

Several approaches to solving the rank deficiency and estimation uncertainty problems have been suggested in the literature:

1. Singular value decomposition of $\mathbf{\Sigma}_d$ provides the linear combinations of the original variables that capture most of the variability in the observations. By retaining less than $n_e - 2$ of these combinations, the EnKF updating procedure is well defined in the reduced space (Skjervheim et al. 2007). Note that exact observations will not be reproduced if this dimensionality reduction is made. Moreover, the procedure contains no direct measure against effects of estimation uncertainty. The approach is expected to work whenever the observations $\mathbf{d}_t$ are highly correlated. Otherwise the information content in the observations can be severely reduced by the dimensionality reduction and the effect of the conditioning may be partly removed.

2. Localization is based on a moving neighborhood linearized conditioning concept. When conditioning a certain dimension of the variables of interest, $\mathbf{x}^u$, only observations in some predefined neighborhood of this dimension is included in the conditioning expression. The conditioning is performed by running sequentially through the dimensions of $\mathbf{x}^u$ with different neighborhood for each dimension, (Evensen 2007; Houtekamer and Mitchell 1998). This reduction of number of conditioning observations for each dimension of $\mathbf{x}^u$ will also reduce the actual dimensions of the $\mathbf{\Gamma}_{x,d}\mathbf{\Sigma}_d^{-1}$-term and hence by adjusting the neighborhood definition full rank for a given ensemble size n_e can be ensured. The approach is expected to work whenever the observations in $\mathbf{d}_t$ are close to

conditional independent given $\mathbf{x}_t$ and each observation has local influence on $\mathbf{x}_t$. Then reliable neighborhood rules can be defined. If this local dependence is not present, the information content in the observations may be severely reduced and the effect of the conditioning will be partly removed. Moreover, localization often introduces artifacts in the solutions since different sets of conditioning observations are used for different dimensions of $\mathbf{x}^u$.

3. Regularization approaches when estimating $\boldsymbol{\Gamma}_{x,d}$ and $\boldsymbol{\Sigma}_d$ have been proposed. The estimates $\widehat{\boldsymbol{\Gamma}}_{x,d}$ and $\widehat{\boldsymbol{\Sigma}}_d^{-1}$ have to be valid covariance matrices and hence non-negative definite. This constrains the class of possible estimators. Non-negative definiteness is closed under addition and certain products, however, and these properties are used to introduce subjective shrinkage effects into the estimator (Hamill et al. 2001). Regularization can both ensure full rank and dampen the effect of large estimation uncertainty. It can of course be complicated to choose a suitable regularization factor and for many proposed regularization factors the consistency of the estimators as $n_e \to \infty$ is lost.

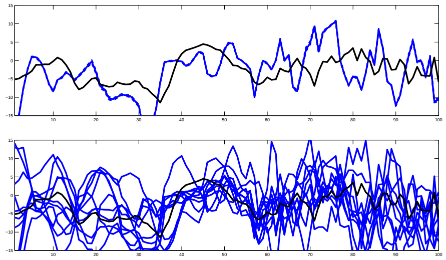
4. Experimental design approaches in generating the initial ensemble has been proposed to reduce the variance of $\widehat{\boldsymbol{\Gamma}}_{x,d}$ and $\widehat{\boldsymbol{\Sigma}}_d^{-1}$ at later times (Evensen 2004). By designing the initial ensemble the iid properties are lost and it is unclear how this influences the bias of the estimates and the final inference of $f(\mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T)$ based on the ensemble $\mathbf{e}_{T+1}$. Experimental design approaches will have no impact on the rank deficiency problem.

5. Ensemble splitting has been proposed in order to explore the estimation uncertainty and reduce the unwanted dependencies between the ensemble members (Houtekamer and Mitchell 1998). By splitting the ensemble the rank deficiency problem becomes even more acute and estimation uncertainty is even worse for each split ensemble. The effect of these features is unclear.

6. A hierarchical extension of the hidden Markov process such that the model parameters $\boldsymbol{\mu}_{xd}$ and $\boldsymbol{\Sigma}_{xd}$ are included in the probability space has been proposed to account for the parameter uncertainty in the EnKF results (Myrseth and Omre 2008). This approach also ensures full rank and provides shrinkage estimates of $\boldsymbol{\Gamma}_{x,d}$ and $\boldsymbol{\Sigma}_d$. This hierarchical EnKF requires that prior pdfs be defined for $\boldsymbol{\mu}_{xd}$ and $\boldsymbol{\Sigma}_{xd}$, which of course can be a challenge. Full consistency in the estimator as $n_e \to \infty$ is ensured, however.

Performance on the Test Examples

The EnKF with the ensemble size $n_e = 30$ is run on the linear test problem to illustrate the effect of estimation uncertainty. Note that rank deficiency is avoided by ensuring $n_e > p_d + 2$. The solutions are presented in Figure 1.6 in a format similar to Figure 1.5. Figure 1.6 should be compared to Figure 1.5 where solutions from the same EnKF algorithm with $n_e = 100$ are presented. Note that the prediction interval based on a single ensemble is severely underestimated while almost all variability is



Linear case

Figure 1.6 Results of the EnKF algorithm with $n_e = 30$ on the linear case: reference realization $\mathbf{x}_{11}$ (black), prediction (blue) and empirical .95-prediction intervals (hatched blue).

shifted to differences between predictions of repeated EnKF runs. The results from the EnKF with a small number of ensemble members seems to be very unreliable.

1.4.3 A Special Case

We consider a hidden Markov processes with nonlinear prior model and Gauss-linear likelihood model in particular, since this case is frequently used in the literature (Evensen 2007). The model assumptions are:

$$\begin{aligned} \mathbf{x}_0 &\sim f(\mathbf{x}_0) \\ [\mathbf{x}_{t+1}|\mathbf{x}_t] &= \omega_t(\mathbf{x}_t, \boldsymbol{\epsilon}_t^x) \sim f(\mathbf{x}_{t+1}|\mathbf{x}_t) \\ [\mathbf{d}_t|\mathbf{x}_t] &= \mathbf{H}_t\mathbf{x}_t + \boldsymbol{\epsilon}_t^d \sim f(\mathbf{d}_t|\mathbf{x}_t) = N_{p_d}(\mathbf{H}_t\mathbf{x}_t, \boldsymbol{\Sigma}_t^d), \end{aligned}$$

where the notation is identical to that in Expressions (1.1) and (1.2). The known forward model $\omega_t(\mathbf{x}_t, \boldsymbol{\epsilon}_t^x)$ may include a differential equation, and it may be extremely computationally demanding to evaluate. Under these model assumptions the EnKF algorithm requires the following ensembles:

$$\mathbf{e}_t \; : \; \{\mathbf{x}_t^{u(i)} \, ; \, i = 1, ..., n_e\} \, ; \, t = 0, ..., T+1,$$

where $\mathbf{x}_t^{u(i)} = [\mathbf{x}_t|\mathbf{d}_0, ..., \mathbf{d}_{t-1}]^{(i)}$ are approximately independent realizations from the conditional distribution $f(\mathbf{x}_t|\mathbf{d}_0, ..., \mathbf{d}_{t-1})$. At each step the associated expectation vector and covariance matrix are $\boldsymbol{\mu}_x$ and $\boldsymbol{\Sigma}_x$ with time reference t omitted. Note that the ensemble need not contain the realizations of observations since the associated $\boldsymbol{\Gamma}_{x,d} = \boldsymbol{\Sigma}_x\mathbf{H}'$ can be assessed from estimates of $\boldsymbol{\Sigma}_x$. The actual EnKF algorithm is as follows:

Algorithm 4: Ensemble Kalman filter (Gauss-linear likelihood)

- Initiate

  $n_e =$ no. of ensemble members
  $\mathbf{x}_0^{u(i)}$; $i = 1, ..., n_e$ iid $f(\mathbf{x}_0)$
  $\mathbf{e}_0 \,:\, \{\mathbf{x}_0^{u(i)}\,;\, i = 1, ..., n_e\}$

- Iterate $t = 0, ..., T$

  Conditioning:
  Estimate $\mathbf{\Sigma}_x$ from $\mathbf{e}_t \rightarrow \widehat{\mathbf{\Sigma}}_x$
  $\mathbf{d}_t^{(i)} \sim N_{p_d}(\mathbf{H}_t\mathbf{x}_t^{u(i)}, \mathbf{\Sigma}_t^d)\,;\, i = 1, ..., n_e$
  $\mathbf{x}_t^{c(i)} = \mathbf{x}_t^{u(i)} + \widehat{\mathbf{\Sigma}}_x\mathbf{H}_t'[\mathbf{H}_t\widehat{\mathbf{\Sigma}}_x\mathbf{H}_t' + \mathbf{\Sigma}_t^d]^{-1}(\mathbf{d}_t - \mathbf{d}_t^{(i)})\,;\, i = 1, ..., n_e$
  Forwarding:
  $\boldsymbol{\epsilon}_t^{x(i)} \sim N_{p_x}(\mathbf{0}, \mathbf{I}_{p_x})\,;\, i = 1, ..., n_e$
  $\mathbf{x}_{t+1}^{u(i)} = \omega_t(\mathbf{x}_t^{c(i)}, \boldsymbol{\epsilon}_t^{x(i)})\,;\, i = 1, ..., n_e$
  $\mathbf{e}_{t+1} \,:\, \{\mathbf{x}_{t+1}^{u(i)}\,;\, i = 1, ..., n_e\}$

- end iterate

- Assess

  $f(\mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T)$ from $\mathbf{e}_{T+1}$

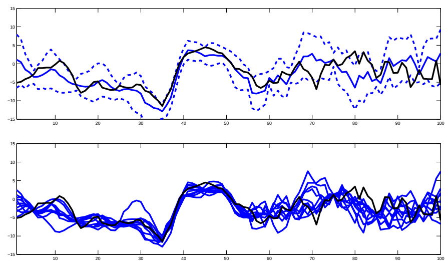
The resulting ensemble $\mathbf{e}_{T+1}$ contains approximately independent realizations of

$$\mathbf{x}_{T+1}^u = [\mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T]$$

from $f(\mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T)$; estimates of the forecast expectation vector and covariance matrix can be obtained as in Expression (1.4). Note that for linear conditioning to be exact, $f(\mathbf{x}_t^u)$ needs to be Gaussian and $f(\mathbf{d}_t|\mathbf{x}_t)$ Gauss-linear. The former need not be the case under the current model assumptions, hence only approximations are obtained.

Performance on the Test Examples

The EnKF-Special case requires the likelihood model to be Gauss-linear and known, hence it can only be applied to the linear case. The solutions are presented in Figure 1.7 together with the $\mathbf{x}_{11}$ reference realization. The EnKF-Special case is based on the ensemble size $n_e = 100$. The layout of the figure is similar to that in Figure 1.5. Figure 1.7 should be compared to Figure 1.3, which displays the correct KF-solution, and Figure 1.5, which displays the solution for the general EnKF algorithm. The latter does estimate the likelihood model. The EnKF-special case with the likelihood model given and $n_e = 100$ provides results that are very similar to the correct KF



Linear case

Figure 1.7 Results of the EnKF-special case algorithm applied to the linear case: reference realization $\mathbf{x}_{11}$ (black), prediction (blue) and empirical .95-prediction intervals (hatched blue).

solution. For the linear case, the EnKF-Special case solution appears more reliable than that of the general algorithm. There is obviously a cost in estimating the likelihood model. Note, however, that in the nonlinear case only the general EnKF can be used.

1.5 The Randomized Maximum Likelihood Filter (RMLF)

The basic idea of the RMLF is very similar to that of the EnKF, except for a different conditioning procedure for the observations. Realizations from the initial model, so called ensemble, are iteratively adjusted for observations and forwarded in time. At time $t = T + 1$, a set of approximately independent realizations is available for assessment of $f(\mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T)$. Basic references for RMLF, also termed iterative EnKF, are Kitanidis (1986); Oliver (1996); Reynolds et al. (2006). The RMLF is well defined on a model for the hidden Markov process with a slightly constrained likelihood model:

$$\begin{aligned} \mathbf{x}_0 &\sim f(\mathbf{x}_0) \\ [\mathbf{x}_{t+1}|\mathbf{x}_t] &= \omega_t(\mathbf{x}_t, \boldsymbol{\epsilon}_t^x) \sim f(\mathbf{x}_{t+1}|\mathbf{x}_t) \\ [\mathbf{d}_t|\mathbf{x}_t] &= \nu_t(\mathbf{x}_t) + \boldsymbol{\epsilon}_t^d \sim f(\mathbf{d}_t|\mathbf{x}_t) = N_{p_d}(\nu_t(\mathbf{x}_t), \boldsymbol{\Sigma}_t^d), \end{aligned}$$

where the notation is identical to that in Expression (1.3). The likelihood model is still nonlinear in $\mathbf{x}_t$, but it is constrained to have an additive Gaussian error term independent of $\mathbf{x}_t$ with known covariance matrix.

A time series of ensembles is defined:

$$\mathbf{e}_t \,:\, \{\mathbf{x}_t^{u(i)}\,;\, i = 1, ..., n_e\}\,;\, t = 0, ..., T + 1,$$

where

$$\mathbf{x}_t^{u(i)} = [\mathbf{x}_t|\mathbf{d}_0, ..., \mathbf{d}_{t-1}]^{(i)}$$

are approximately independent realizations from $f(\mathbf{x}_t|\mathbf{d}_0, ..., \mathbf{d}_{t-1})$. At any time step the associated expectation vector and covariance matrix are $\boldsymbol{\mu}_x$ and $\boldsymbol{\Sigma}_x$ with time reference t omitted. The RMLF algorithm in its simplest form is included here to illustrate the basic idea of the approach:

Algorithm 5: Randomized maximum likelihood filter

- Initiate

  $n_e =$ no. of ensemble members
  $\mathbf{x}_0^{u(i)}\,;\, i = 1, ..., n_e$ iid $f(\mathbf{x}_0)$
  $\mathbf{e}_0\,:\,\{\mathbf{x}_0^{u(i)}\,;\, i = 1, ..., n_e\}$

- Iterate $t = 0, ..., T$

  Conditioning:
  Estimate $\boldsymbol{\Sigma}_x$ from $\mathbf{e}_t \rightarrow \widehat{\boldsymbol{\Sigma}}_x$
  $\mathbf{o}_t^{(i)} \sim N_{p_d}(\mathbf{d}_t, \boldsymbol{\Sigma}_t^d)\,;\, i = 1, ..., n_e$
  $$\mathbf{x}_t^{c(i)} = \mathrm{argmin}_{\mathbf{x}}\{(\mathbf{o}_t^{(i)} - \nu_t(\mathbf{x}))'\boldsymbol{\Sigma}_t^{d\,-1}(\mathbf{o}_t^{(i)} - \nu_t(\mathbf{x})) + (\mathbf{x} - \mathbf{x}_t^{u(i)})'\widehat{\boldsymbol{\Sigma}}_x^{-1}(\mathbf{x} - \mathbf{x}_t^{u(i)})\}\,;\, i = 1, ..., n_e$$
  Forwarding:
  $\boldsymbol{\epsilon}_t^{x(i)} \sim N_{p_x}(\mathbf{0}, \mathbf{I}_{p_x})\,;\, i = 1, ..., n_e$
  $\mathbf{x}_{t+1}^{u(i)} = \omega_t(\mathbf{x}_t^{c(i)}, \boldsymbol{\epsilon}_t^{x(i)})\,;\, i = 1, ..., n_e$
  $\mathbf{e}_{t+1}\,:\,\{\mathbf{x}_{t+1}^{u(i)}\,;\, i = 1, ..., n_e\}$

- end iterate

- Assess

  $f(\mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T)$ from $\mathbf{e}_{T+1}$

The resulting ensemble $\mathbf{e}_{T+1}$ contains approximately independent realizations of

$$\mathbf{x}_{T+1}^u = [\mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T]$$

from $f(\mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T)$. Hence estimates of the forecast expectation vector and covariance matrix can be obtained with the usual estimators as in Expression 1.4. The RMLF algorithm, Algorithm 5, is also recursive with a conditioning operation and a forwarding operation in each step. Note that $\mathbf{o}_t^{(i)}$ is not identical to $\mathbf{d}_t^{(i)}$ used in EnKF, since the former is centered around the actual observation $\mathbf{d}_t$. There are two implicit approximations in the RMLF:

1. Discretization of the sample space of $\mathbf{x}_t$. The initial ensemble is assumed to represent $f(\mathbf{x}_0)$. This approximation is similar for all ensemble based approaches and it may require a large number of ensemble members in high-dimensional problems.

2. Conditioning on observations $\mathbf{d}_t$ is done by optimization with a criterion that relies on Gaussianity. This optimization approach is more flexible than the linearization used in the EnKF. Note that each ensemble member is conditioned individually, hence the approximation only applies to the conditioning, not to the forward model. For highly non-Gaussian prior models this approximation may provide unreliable solutions.

Under these approximations, however, all hidden Markov processes with nonlinear likelihood models and additive Gaussian error terms can be evaluated. The RMLF coincides with the EnKF whenever the likelihood model is Gauss-linear (see Appendix B). This entails that it is a consistent forecast procedure under a full Gauss-linear model in the sense that the exact solution is obtained as $n_e \to \infty$ (see Appendix A).

Two major problems that may arise with the RMLF are:

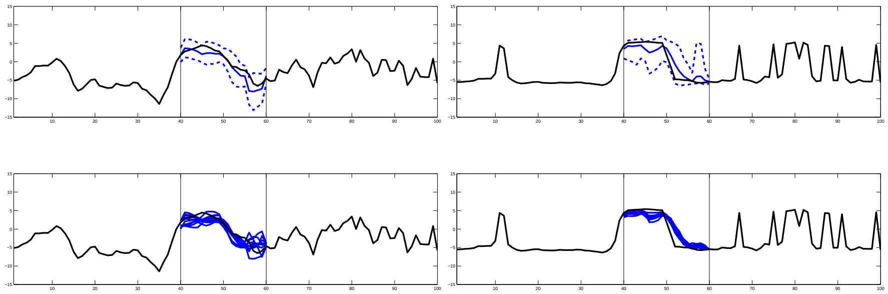
1. Complications caused by the use of an estimate of $\mathbf{\Sigma}_x$ based on the ensemble $\mathbf{e}_t$ instead of the true covariance matrix. These complications include rank deficiency and estimation uncertainty as in EnKF.

2. Computational cost of performing the optimization in each step; specially because the objective function may have several local optima. If the nonlinear likelihood function $\nu_t(.)$ is available in analytic form so that the derivatives are computable, an efficient optimization may be performed. For a general 'black box' $\nu_t(.)$ used in a computer code, the optimization may be prohibitively expensive to perform.

The RMLF is not widely used but several applications to reservoir evaluation problems demonstrate its characteristics (Gao et al. 2006; Gu and Oliver 2007). It is shown, however, that the conditioning procedure in the RMLF captures multimodality in the prior model and likelihood function better than the EnKF.

Performance on the Test Examples

The RMLF approach can be applied to both the linear and nonlinear cases, but due to the high computational cost of the optimization, only the short section $[40, 60]$ is evaluated, see Figure 1.8. For the linear case the RMLF and the EnKF-Special case solution in Figure 1.7 coincide. All solutions are trustworthy with the ensemble size $n_e = 100$. In the nonlinear case the RMLF solution can be compared to the EnKF solution in Figure 1.5, and the former appears more reliable since the prediction intervals look more trustworthy and there is less variability between repeated runs. Computational demands may limit the use of the RMLF, however.



Linear case Nonlinear case

Figure 1.8 Results of the RMLF algorithm on the linear and nonlinear test cases: reference realization $\mathbf{x}_{11}$ (black), prediction (blue) and empirical .95-prediction intervals (hatched blue).

1.6 The Particle Filter (PF)

The basic idea of the particle filter is to create a set of iid realizations from the initial model, so called particles, and to take these realizations through the prior model. Each realization is assigned weights according to the likelihood model as observations occur through time. At time $t = T+1$, a set of realizations with associated weights are available for assessment of $f(\mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T)$. Note that the actual ensemble members of $\mathbf{e}_{T+1}$, without weighting, carry no information about the observations, hence visual inspection is of no use. Basic references for particle filters are Gordon et al. (1993) and Doucet et al. (2001).

A time series of ensembles is defined:

$$\mathbf{e}_t \,:\, \{(\mathbf{x}_t, w_t)^{(i)}; i = 1, ..., n_e\}\,;\, t = 0, ..., T+1$$

where $\mathbf{x}_t^{(i)}$ are samples from the prior model of $\mathbf{x}_t$, and $w_t^{(i)}$ are associated weights that are updated as observations arise. The particle filter algorithm in its simplest form is as follows:

Algorithm 6: Particle filter

- Initiate:

$$\begin{aligned}
&n_e = \text{ no. of ensemble members}\\
&\mathbf{x}_0^{(i)}\,;\, i = 1, ..., n_e \text{ iid } f(\mathbf{x}_0)\\
&w_0^{(i)} = 1\,;\, i = 1, ..., n_e\\
&\mathbf{e}_0 \,:\, \{(\mathbf{x}_0, w_0)^{(i)}\,;\, i = 1, ..., n_e\}
\end{aligned}$$

- Iterate $t = 0, ..., T$

Weight updating:
$v^{(i)} = f(\mathbf{d}_t|\mathbf{x}_t^{(i)}) \times w_t^{(i)}\ ; \ i = 1, ..., n_e$
$w_{t+1}^{(i)} = v^{(i)} \times [\sum_{i=1}^{n_e} v^{(i)}]^{-1}\ ; \ i = 1, ..., n_e$
Forwarding:
$\boldsymbol{\epsilon}_t^{x(i)} \sim N_{p_x}(\mathbf{0}, \mathbf{I}_{p_x})\ ; \ i = 1, ..., n_e$
$\mathbf{x}_{t+1}^{u(i)} = \omega_t(\mathbf{x}_t^{c(i)}, \boldsymbol{\epsilon}_t^{x(i)})\ ; \ i = 1, ..., n_e$
$\mathbf{e}_{t+1}\ :\ \{(\mathbf{x}_{t+1}, w_{t+1})^{(i)}\ ; \ i = 1, ..., n_e\}$

- end iterate
- Assess

$f(\mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T)$ from $\mathbf{e}_{T+1}$

The resulting ensemble $\mathbf{e}_{T+1}$ contains samples from the prior model $f(\mathbf{x}_{T+1})$ and associated normalized weights resulting from conditioning on $[\mathbf{d}_0, ..., \mathbf{d}_T]$ which can be used in assessing the forecast pdf $f(\mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T)$. Estimates of the forecasting expectation vector and covariance matrix will be:

$$\widehat{\boldsymbol{\mu}}_{T+1} = \widehat{\mathrm{E}}\{\mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T\} = \sum_{i=1}^{n_e} w_{T+1}^{(i)} \mathbf{x}_{T+1}^{(i)}$$

$$\widehat{\boldsymbol{\Sigma}}_{T+1} = \widehat{\mathrm{Var}}\{\mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T\} = \sum_{i=1}^{n_e} w_{T+1}^{(i)} (\mathbf{x}_{T+1}^{(i)} - \widehat{\boldsymbol{\mu}}_{T+1})(\mathbf{x}_{T+1}^{(i)} - \widehat{\boldsymbol{\mu}}_{T+1})'.$$

The PF algorithm, Algorithm 6, is recursive with a weight updating operation and a forwarding operation in each step. The implicit approximation in the particle filter lies in the discretization of the sample space of $\mathbf{x}_t$. The initial ensemble is assumed to represent $f(\mathbf{x}_0)$. For high-dimensional problems a large number of ensemble members may be required to justify these assumptions. Under this approximation, however, all types of models for the hidden Markov process can be evaluated. The particle filter is a consistent forecast procedure for all hidden Markov models in the sense that the exact solution is obtained as $n_e \to \infty$.

The particle filter can be run efficiently but one major problem is that all weights often are assigned to one or few samples. This happens when the prior model is vague and/or the observations are very informative. Several ways of correcting this problem have been proposed (Doucet et al. 2001). Since particle filters are not the focus of this presentation we leave the reader with this reference and the following example.

Performance on the Test Examples

The particle filter can be applied to both the linear and nonlinear cases, see Figure 1.9. The filter is computationally efficient and several hundred thousand ensemble

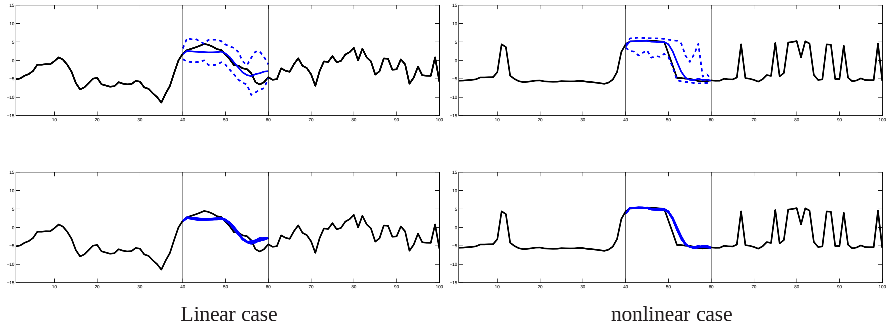


Figure 1.9 Results of the particle filter algorithm on the linear and nonlinear cases: reference realization $\mathbf{x}_{11}$ (black), prediction (blue) and empirical .95-prediction intervals (hatched blue).

members were run. Even with so many ensemble members only a few members were assigned weights significantly different from zero for the full scale example, however. Hence we limit the study to the section $[40, 60]$ and used the ensemble size $n_e = 130\,000$, which corresponds to an effective number of ensemble members (Doucet et al. 2001) around 100. The particle filter results for the linear example should be compared to the exact solution obtained with the KF in Figure 1.3. It appears that the PF solution is close to the KF solution. For the nonlinear case the particle filter results should be compared to the EnKF and RMLF solutions in Figures 1.5 and 1.8, respectively. The PF solution appears reliable and the prediction interval seems to be close to the RMLF solution. The results indicate that the PF needs extremely many ensemble members in order to provide reliable solutions.

1.7 Closing Remarks

The forecasting of $[\mathbf{x}_{T+1}|\mathbf{d}_0, ..., \mathbf{d}_T]$ is based on a hidden Markov model. If this model is assumed to be Gauss-linear it is subject to analytical evaluation and traditional KF provides the exact solution. With deviations from Gauss-linearity approximate solutions must be sought, and a large variety of approaches are available. In the current paper focus is on simulation based inference of the hidden Markov model, and the EnKF in particular. Also RMLF and PF are being introduced.

The content of this paper can be summarized by:

- The EnKF and RMLF on one hand and PF on the other, are based on very different ideas, although both rely on simulation based inference. The EnKF and RMLF are based on an ensemble where each ensemble member is adjusted as observations appear. Hence the ensemble members are approximately independent realizations of $[\mathbf{x}_t|\mathbf{d}_0, ..., \mathbf{d}_{t-1}]$ at any time. The PF, on the other hand, is based on an ensemble where each ensemble member is fixed but weighted. The weights are sequentially adjusted as observations appear. Hence the weighted ensemble members represent $f(\mathbf{x}_t|\mathbf{d}_0, ..., \mathbf{d}_{t-1})$. In practise, one can visualize the ensemble members of the EnKF and the RMLF to illustrate the solution, but doing so for the PF ensemble has no meaning.

- The EnKF can be used for general hidden Markov models. It relies on a linearization of the conditioning expression where the weights are estimates from the ensemble. Hence EnKF will only provide an approximate solution even when the ensemble size tends to infinity. For high-dimensional problems the estimated weights may be unreliable if the ensemble size is too small. The EnKF is relatively computationally efficient.

- The RMLF can be used for general hidden Markov models, with minor constraints on the likelihood model. The conditioning expression is phrased as an optimization with an object function inferred from the ensemble. The RMLF will provide more reliable results than the EnKF whenever there are nonlinearities in the model. The solutions will only be approximate, however, even when the ensemble size tends to infinity. The RMLF may be very computationally demanding for high-dimensional problems with object functions with multiple optima.

- The PF can also be used for general hidden Markov models. The conditioning is made exactly, and the correct solution will be obtained whenever the ensemble size tends to infinity. Moreover, the PF is extremely computationally efficient. For high dimensional problems however, the convergence towards the correct solution is extremely slow. In practice, only very few ensemble members are assigned weight which are significantly different from zero.

The authors general recommendations are: For very low dimensional problem the PF is to be recommended due to its favorable asymptotic properties. For intermediate and high dimensional problems the EnKF is expected to perform better than the PF given comparable computer resources. The adjustment of the individual ensemble members in the EnKF provides better coverage in high probability areas of the sample space.Attention must be given to the reliability of the EnKF solution, however, due to possible lack of precision in the estimated linearization. The RMLF is expected to find its use in special intermediate-dimensional problems with nonlinear relations.

Bibliography

Bertino L, Evensen G and H. W 2002 Combining geostatistics and Kalman filtering for data assimilation in an estuarine system. *Inverse Problems* **18**(1), 1–23.

Bertino L, Evensen G and Wackernagel H 2003 Sequential data assimilation techniques in oceanography. *International Statistical Review* **71**, 223–241.

Burgers G, van Leeuwen PJ and Evensen G 1998 Analysis scheme in the ensemble Kalman filter. *Monthly Weather Review* **126**(6), 1719–1724.

Dovera L and Della Rossa L 2007 Ensemble Kalman filters for Gaussian mixture models *In Extended abstracts book of Petroleum Geostatistics 2007, Cascais, Portugal*.

(ed. Doucet A, de Freitas N and Gordon N) 2001 *Sequential Monte Carlo Methods in Practice*. Springer-Verlag.

Evensen G 1994 Sequential data assimilation with nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *Journal of Geophysical Research* **99**(C5), 10,143–10,162.

Evensen G 2004 Sampling strategies and square root analysis schemes for the EnKF. *Ocean Dynamics* **54**(6), 539–560.

Evensen G 2007 *Data Assimilation; The Ensemble Kalman Filter*. Springer.

Gao G, Zafari M and Reynolds AC 2006 Quantifying uncertainty for the PUNQ-S3 problem in a Bayesian setting with RML and EnKF. *SPE Journal* **11**(4), 506–515.

Gordon NJ, Salmond DJ and Smith AFM 1993 Novel approach to nonlinear/non-Gaussian Bayesian state estimation *In IEE Proceedings-F*, vol. 140, pp. 107–113.

Gu Y and Oliver DS 2007 An iterative ensemble Kalman filter for multiphase fluid flow data assimilation. *SPE Journal* **12**(4), 438–446.

Hamill TM, Whitaker JS and Snyder C 2001 Distance-dependent filtering of background error covariance estimates in an ensemble Kalman filter. *Monthly Weather Review* **129**(11), 2776–2790.

Houtekamer P and Mitchell HL 1998 Data assimilation using an ensemble Kalman filter technique. *Monthly Weather Review* **126**(3), 796–811.

Houtekamer P, Mitchell HL, Pellerin G, Buehner M, Charron M, Spacek L and Hansen B 2005 Atmospheric data assimilation with an ensemble Kalman filter: Results with real observations. *Monthly Weather Review* **133**(3), 604–620.

Huber PJ 1981 *Robust Statistics*. Wiley.

Jafarpour B and McLaughlin 2007 History matching with an ensemble Kalman filter and discrete cosine parameterization *In SPE Annual Technical Conference and Exhibition, 11-14 November 2007, Anaheim, California, U.S.A.*

Jazwinski AH 1970 *Stochastic processes for filtering theory*. Academic Press.

Julier SJ and Uhlmann JK 1997 A new extension of the Kalman filter to nonlinear systems *In proceedings of the 12th International Symposium on Aerospace/Defense Sensing, Simulation and Controls*, pp. 182–193.

Kalman RE 1960 A new approach to linear filtering and prediction problems. *Journal of Basic Engineering* **82**(1), 35–45.

Kelker D 1970 Distribution theory of spherical distributions and a location-scale parameter generalization. *Sankhya: The Indian Journal of Statistics* **32**, 419–430.

Kitanidis PK 1986 Parameter uncertainty in estimation of spatial functions: Bayesian estimation. *Water resources research* **22**(4), 499–507.

Liu N and Oliver DS 2005a Critical evaluation of the ensemble Kalman filter on history matching of geologic facies. *SPE Reservoir Evaluation & Engineering* **8**, 470–477.

Liu N and Oliver DS 2005b Ensemble Kalman filter for automatic history matching of geologic facies. *Journal of Petroleum Science and Engineering* **47**(3), 147–161.

Myrseth I and Omre H 2008 Hierarchical ensemble Kalman filter. *To be submitted*.

Nævdal G, Johnsen LM, Aanonsen SI and Vefring EH 2005 Reservoir monitoring and continuous model updating using ensemble Kalman filter. *SPE Journal* **10**(1), 66–74.

Oliver DS 1996 On conditional simulation to inaccurate data. *Journal of Mathematical Geology* **28**(6), 811–817.

Reynolds AC, Zafari M and Li G 2006 Iterative forms of the ensemble Kalman filter *In Proceedings of the 10th European conference on the Mathematics of Oil Recovery*, pp. 182–193.

Skjervheim J, Evensen G, Aanonsen S, Ruud B and Johansen T 2007 Incorporating 4D seismic data in reservoir simulation models using ensemble Kalman filter. *SPE Journal* **12**(3), 282–292.

Appendix A Properties of the EnKF Algorithm

We show that the EnKF algorithm is exact in the limit $n_e \to \infty$ under a Gauss-linear model.

The Gauss-linear model is

$$\begin{aligned} f(\mathbf{x}_0) &\sim N_{p_x}(\boldsymbol{\mu}_0^u, \boldsymbol{\Sigma}_0^x) \\ f(\mathbf{x}_{t+1}|\mathbf{x}_t) &\sim N_{p_x}(\mathbf{A}_t\mathbf{x}_t, \boldsymbol{\Sigma}_t^x) \\ f(\mathbf{d}_t|\mathbf{x}_t) &\sim N_{p_d}(\mathbf{H}_t\mathbf{x}_t, \boldsymbol{\Sigma}_t^d). \end{aligned}$$

Note that for consistent estimators and $n_e \to \infty$:

$$\widehat{\boldsymbol{\Sigma}}_{xd} \to \boldsymbol{\Sigma}_{xd} = \begin{bmatrix} \boldsymbol{\Sigma}_x & \boldsymbol{\Gamma}_{x,d} \\ \boldsymbol{\Gamma}_{d,x} & \boldsymbol{\Sigma}_d \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Sigma}_t^u & \boldsymbol{\Sigma}_t^u\mathbf{H}_t' \\ \mathbf{H}_t\boldsymbol{\Sigma}_t^u & \mathbf{H}_t\boldsymbol{\Sigma}_t^u\mathbf{H}_t' + \boldsymbol{\Sigma}_t^d \end{bmatrix}.$$

The proof is done by induction: Assume that

$$\begin{aligned} \mathbf{x}_t^{u(i)} &\sim N_{p_x}(\boldsymbol{\mu}_t^u, \boldsymbol{\Sigma}_t^u) \\ [\mathbf{d}_t^{(i)}|\mathbf{x}_t^{u(i)}] &\sim N_{p_d}(\mathbf{H}_t\mathbf{x}_t^{u(i)}, \boldsymbol{\Sigma}_t^d). \end{aligned}$$

The conditioning rule is:

$$[\mathbf{x}_t^{c(i)}|\mathbf{d}_t^{(i)}, \mathbf{x}_t^{u(i)}] = \mathbf{x}_t^{u(i)} + \boldsymbol{\Gamma}_{x,d}\boldsymbol{\Sigma}_d^{-1}(\mathbf{d}_t - \mathbf{d}_t^{(i)}).$$

Due to linearity and Gaussianity in $[\mathbf{d}_t^{(i)}, \mathbf{x}_t^{u(i)}]$ one has:

$$\mathbf{x}_t^{c(i)} \sim N_{p_x}(\boldsymbol{\mu}_t^c, \boldsymbol{\Sigma}_t^c),$$

with

$$\begin{aligned} \boldsymbol{\mu}_t^c = \mathrm{E}\{\mathbf{x}_t^{c(i)}\} &= \boldsymbol{\mu}_t^u + \boldsymbol{\Gamma}_{x,d}\boldsymbol{\Sigma}_d^{-1}(\mathbf{d}_t - \mathbf{H}_t\boldsymbol{\mu}_t^u) \\ &= \boldsymbol{\mu}_t^u + \boldsymbol{\Sigma}_t^u\mathbf{H}_t'[\mathbf{H}_t\boldsymbol{\Sigma}_t^u\mathbf{H}_t' + \boldsymbol{\Sigma}_t^d]^{-1}(\mathbf{d}_t - \mathbf{H}_t\boldsymbol{\mu}_t^u) \end{aligned}$$

$$\begin{aligned} \boldsymbol{\Sigma}_t^c = \mathrm{Var}\{\mathbf{x}_t^{c(i)}\} &= \boldsymbol{\Sigma}_t^u + \boldsymbol{\Gamma}_{x,d}\boldsymbol{\Sigma}_d^{-1}[\mathbf{H}_t\boldsymbol{\Sigma}_t^u\mathbf{H}_t' + \boldsymbol{\Sigma}_t^d]\boldsymbol{\Sigma}_d^{-1}\boldsymbol{\Gamma}_{d,x} \\ &- 2\boldsymbol{\Gamma}_{x,d}\boldsymbol{\Sigma}_d^{-1}\mathbf{H}_t'\boldsymbol{\Sigma}_t^u \\ &= \boldsymbol{\Sigma}_t^u + \boldsymbol{\Sigma}_t^u\mathbf{H}_t'[\mathbf{H}_t\boldsymbol{\Sigma}_t^u\mathbf{H}_t' + \boldsymbol{\Sigma}_t^d]^{-1}\mathbf{H}_t\boldsymbol{\Sigma}_t^u \\ &- 2\boldsymbol{\Sigma}_t^u\mathbf{H}_t'[\mathbf{H}_t\boldsymbol{\Sigma}_t^u\mathbf{H}_t' + \boldsymbol{\Sigma}_t^d]^{-1}\mathbf{H}_t\boldsymbol{\Sigma}_t^u \\ &= \boldsymbol{\Sigma}_t^u - \boldsymbol{\Sigma}_t^u\mathbf{H}_t'[\mathbf{H}_t\boldsymbol{\Sigma}_t^u\mathbf{H}_t' + \boldsymbol{\Sigma}_t^d]^{-1}\mathbf{H}_t\boldsymbol{\Sigma}_t^u, \end{aligned}$$

which corresponds to the KF solution in Algorithm 2.

Moreover:

$$[\mathbf{x}_{t+1}^{u(i)}|\mathbf{x}_t^{c(i)}] \sim N_{p_x}(\mathbf{A}_t\mathbf{x}_t^{c(i)}, \mathbf{\Sigma}_t^x)$$
$$\mathbf{x}_{t+1}^{u(i)} \sim N_{p_x}(\mathbf{A}_t\boldsymbol{\mu}_t^c, \mathbf{A}_t\mathbf{\Sigma}_t^c\mathbf{A}_t' + \mathbf{\Sigma}_t^x),$$

which corresponds to the KF solution.

Since Gauss-linearity entails that for $t = 0$:

$$\mathbf{x}_0^{u(i)} \sim N_{p_x}(\boldsymbol{\mu}_0^x, \mathbf{\Sigma}_0^x)$$
$$[\mathbf{d}_0^{(i)}|\mathbf{x}_0^{u(i)}] \sim N_{p_d}(\mathbf{H}_0\mathbf{x}_0^{u(i)}, \mathbf{\Sigma}_0^d)$$

it is concluded by induction that all ensemble members in EnKF under a Gauss-linear model are sampled from the correct pdf. QED.

Appendix B Properties of the RMLF Algorithm

We show that the RMLF algorithm coincides with the EnKF algorithm whenever the likelihood model is Gauss-linear.

The Gauss-linear likelihood model:

$$\mathbf{o}_t^{(i)} \sim N_{p_d}(\mathbf{d}_t, \mathbf{\Sigma}_t^d)$$
$$f_r^{(i)}(\mathbf{x}) = N_{p_x}(\mathbf{x}_t^{u(i)}, \widehat{\mathbf{\Sigma}}_x)$$
$$f_r^{(i)}(\mathbf{o}_t^{(i)}|\mathbf{x}) = N_{p_d}(\mathbf{H}_t\mathbf{x}, \mathbf{\Sigma}_t^d).$$

The conditioning rule with $f^{(i)}(\mathbf{x}|\mathbf{o}_t^{(i)})$ being Gaussian is:

$$\begin{aligned}
\mathbf{x}_t^{c(i)} &= \mathrm{argmin}_{\mathbf{x}}\{(\mathbf{o}_t^{(i)} - \mathbf{H}_t\mathbf{x})\mathbf{\Sigma}_t^{d^{-1}}(\mathbf{o}_t^{(i)} - \mathbf{H}_t\mathbf{x})' + (\mathbf{x} - \mathbf{x}_t^{u(i)})\widehat{\mathbf{\Sigma}}_x^{-1}(\mathbf{x} - \mathbf{x}_t^{u(i)})'\} \\
&= \mathrm{argmax}_{\mathbf{x}}\{f_r^{(i)}(\mathbf{o}_t^{(i)}|\mathbf{x})f_r^{(i)}(\mathbf{x})\} \\
&= \mathrm{argmax}_{\mathbf{x}}\{f^{(i)}(\mathbf{x}|\mathbf{o}_t^{(i)})\} \\
&= \mathrm{MAP}\{\mathbf{x}|\mathbf{o}_t^{(i)}, \mathbf{x}_t^{u(i)}\} \\
&= \mathrm{E}\{\mathbf{x}|\mathbf{o}_t^{(i)}, \mathbf{x}_t^{u(i)}\}.
\end{aligned}$$

Hence a better name would have been Randomized maximum aposteriori filter (RMAPF).

Hereby

$$\begin{aligned}
\mathbf{x}_t^{c(i)} &= \mathbf{x}_t^{u(i)} + \widehat{\mathbf{\Sigma}}_x\mathbf{H}_t'\mathbf{\Sigma}_o^{-1}[\mathbf{o}_t^{(i)} - \mathbf{H}_t\mathbf{x}_t^{u(i)}] \\
&= \mathbf{x}_t^{u(i)} + \widehat{\mathbf{\Sigma}}_x\mathbf{H}_t'[\mathbf{H}_t\widehat{\mathbf{\Sigma}}_x\mathbf{H}_t' + \mathbf{\Sigma}_t^d]^{-1}(\mathbf{d}_t - \mathbf{d}_t^{(i)}),
\end{aligned}$$

with

$$\mathbf{d}_t = \mathbf{o}_t^{(i)} - \boldsymbol{\epsilon}_t^d$$
$$\mathbf{d}_t^{(i)} = \mathbf{H}_t \mathbf{x}_t^{u(i)} + \boldsymbol{\epsilon}_t^d,$$

which corresponds to the EnKF conditioning rule in Algorithm 4. QED.