Resampling the Ensemble Kalman Filter

Inge Myrseth, Jon Sætrom and **Henning Omre,**
Norwegian University of Science and Technology

Summary

Ensemble Kalman filters (EnKF) based on a small ensemble tend to provide collapse of the ensemble over time. It is shown that this collapse is caused by positive coupling of the ensemble members due to use of one common estimate of the Kalman gain for the update of all ensemble members at each time step. This coupling can be avoided by resampling the Kalman gain from its sampling distribution in the conditioning step. In the analytically tractable Gauss-linear model finite sample distributions for all covariance matrix estimates involved in the Kalman gain estimate are known and hence exact Kalman gain resampling can be done. For the general non-linear case we introduce the resampling ensemble Kalman filter (ResEnKF) algorithm. The resampling strategy in the algorithm is based on bootstrapping of the ensemble and Monte Carlo simulation of the likelihood model. An empirical study demonstrates that ResEnKF provides more reliable prediction intervals than traditional EnKF, on the cost of somewhat less accuracy in the point predictions. In a synthetic reservoir study, it is shown the the hierarchical ensemble Kalman filter (HEnKF) provides more reliable predictions and prediction intervals than both ResEnKF and traditional EnKF. HEnKF requires additional modeling, however.

Introduction

The Ensemble Kalman Filter (EnKF) introduced by Evensen in the papers Evensen (1994) and Burgers et al. (1998) has found widespread use in evaluation of spatio-temporal phenomena like ocean modeling, weather forecasting and petroleum reservoir evaluation, see Bertino et al. (2003), Houtekamer et al. (2005), Nævdal et al. (2005) and references therein. The filter is popular because of easy implementation and computational efficiency. The filter relies on simulation based inference of hidden Markov models and is closely related to the traditional Kalman filter. The EnKF utilizes a linearization in the data conditioning and relies on empirical probability densities, represented as an ensemble of possible states, which allow general forward functions. These approximations make the ensemble Kalman filter computationally efficient and well suited for high-dimensional hidden Markov models.

The data conditioning is based on the estimated correlation between observations and ensemble members, which is used to update all ensemble members. The estimated regression weights, so called Kalman gain in the case with linear observation relations, is associated with finite sample uncertainty usually resulting in underestimated Kalman gain, see Furrer and Bengtsson (2007). Anderson (2001) deals with the problem by variance inflation in an effort to maintain variability in the ensemble statistics. One of the key assumptions in the data conditioning is that the ensemble members are independent. However, when using the same estimate of the Kalman gain to update every ensemble member, the ensemble members will be coupled over time.

In the hierarchical ensemble Kalman filter (HEnKF) algorithm (Myrseth and Omre, 2009) uncertainty caused by the Kalman gain estimate is accounted for. The HEnKF algorithm relies on an extended model of the prior on the model parameters, however. In the current paper, we propose to update every ensemble member individually with different estimates of the Kalman gain using a bootstrapping technique (Efron, 1979). The estimation uncertainty associated with the Kalman gain will then be reflected in the ensemble uncertainty. We also introduce formalism that handles non-linear relations between state and observation.

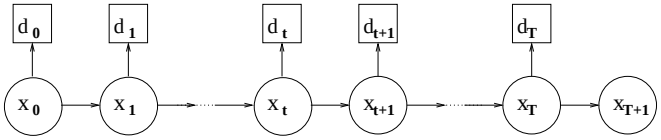

Fig. 1—Hidden Markov process

Model Assumptions

Consider an unknown, multivariate time series $[\mathbf{x}_0, \mathbf{x}_1, ..., \mathbf{x}_T, \mathbf{x}_{T+1}]$ with $\mathbf{x}_t \in \mathbb{R}^{p_\mathbf{x}}$; $t = 0, ..., T+1$ 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 $\mathbf{d}_t \in \mathbb{R}^{p_\mathbf{d}}$; $t = 0, ..., T$, is available.

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

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

Let $f(\mathbf{x}_0)$ be a known pdf for the initial state, and $f(\mathbf{x}_{t+1}|\mathbf{x}_t)$ for $t = 0, ..., T$ 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:

$$\begin{aligned}[\mathbf{d}_0, ..., \mathbf{d}_T | \mathbf{x}_0, ..., \mathbf{x}_{T+1}] &\sim f(\mathbf{d}_0, ..., \mathbf{d}_T | \mathbf{x}_0, ..., \mathbf{x}_{T+1}) \\ &= \prod_{t=0}^{T} f(\mathbf{d}_t | \mathbf{x}_0, ..., \mathbf{x}_{t+1}) \\ &= \prod_{t=0}^{T} f(\mathbf{d}_t | \mathbf{x}_t) \quad \ldots\ldots\ldots\ldots \quad (2)\end{aligned}$$

where $f(\mathbf{d}_t|\mathbf{x}_t)$ for $t = 0, ..., T$ are known likelihood functions. Hence, the likelihood model entails that the observation 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 graph in Figure 1. The resulting posterior stochastic model is defined by Bayesian inversion:

$$\begin{aligned}[\mathbf{x}_0, ..., \mathbf{x}_{T+1} | \mathbf{d}_0, ..., \mathbf{d}_T] &\sim f(\mathbf{x}_0, ..., \mathbf{x}_{T+1} | \mathbf{d}_0, ..., \mathbf{d}_T) \\ &= \text{const} \times f(\mathbf{d}_0, ..., \mathbf{d}_T | \mathbf{x}_0, ..., \mathbf{x}_{T+1}) \\ &\quad \times f(\mathbf{x}_0, ..., \mathbf{x}_{T+1}) \\ &= \text{const} \times f(\mathbf{x}_0) f(\mathbf{d}_0 | \mathbf{x}_0) \\ &\quad \times \left[\prod_{t=0}^{T-1} f(\mathbf{d}_{t+1} | \mathbf{x}_{t+1}) f(\mathbf{x}_{t+1} | \mathbf{x}_t) \right] \\ &\quad \times f(\mathbf{x}_{T+1} | \mathbf{x}_T), \quad \ldots\ldots\ldots\ldots \quad (3)\end{aligned}$$

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

For the hidden Markov model described, the forecast is of inter-

est. The forecasting pdf is available as:

$$[\mathbf{x}_{T+1}|\mathbf{d}_0,...,\mathbf{d}_T] \sim f(\mathbf{x}_{T+1}|\mathbf{d}_0,...,\mathbf{d}_T) = \int ... \int f(\mathbf{x}_0,...,\mathbf{x}_{T+1}|\mathbf{d}_0,...,\mathbf{d}_T)\mathrm{d}\mathbf{x}_0...\mathrm{d}\mathbf{x}_T. \quad (4)$$

This forecasting pdf is computable by a recursive algorithm which alternates a forward-in-time step and a condition-on-data step. This recursive algorithm makes sequential conditioning on future observations possible.

The model described above can be summarized by the following general state space equations:

$$\begin{aligned} \mathbf{x}_0 &\sim f(\mathbf{x}_0) \\ \mathbf{x}_{t+1}|\mathbf{x}_t = \omega_t(\mathbf{x}_t,\varepsilon_t^{\mathbf{x}}) &\sim f(\mathbf{x}_{t+1}|\mathbf{x}_t) \\ \mathbf{d}_t|\mathbf{x}_t = \nu_t(\mathbf{x}_t,\varepsilon_t^{\mathbf{d}}) &\sim f(\mathbf{d}_t|\mathbf{x}_t), \quad \ldots\ldots\ldots\ldots\ldots\ldots (5) \end{aligned}$$

where $\omega_t(.,.)$ is a known function $\mathbb{R}^{2p_{\mathbf{x}}} \to \mathbb{R}^{p_{\mathbf{x}}}$ and $\varepsilon_t^{\mathbf{x}}$ is a random variable from the normalized $p_{\mathbf{x}}$-dimensional multivariate Gaussian distribution $N_{p_{\mathbf{x}}}(\mathbf{0},\mathbf{I}_{p_{\mathbf{x}}})$ where $\mathbf{I}_{p_{\mathbf{x}}}$ is a unit diagonal covariance matrix, $\nu_t(.,.)$ is a known function $\mathbb{R}^{p_{\mathbf{x}}+p_{\mathbf{d}}} \to \mathbb{R}^{p_{\mathbf{d}}}$ and $\varepsilon_t^{\mathbf{d}}$ is a normalized $p_{\mathbf{d}}$-dimensional Gaussian random variable from $N_{p_{\mathbf{d}}}(\mathbf{0},\mathbf{I}_{p_{\mathbf{d}}})$. This construction can generate a realization from an arbitrary forward, $f(\mathbf{x}_{t+1}|\mathbf{x}_t)$, and likelihood, $f(\mathbf{d}_t|\mathbf{x}_t)$, model.

The Ensemble Kalman Filter

The EnKF is an algorithm that can be used to assess the forecasting pdf. The basic idea of the EnKF to represent an empirical distribution approximating the true prior by a set of realizations, so called ensemble. 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. Basic references for EnKF are Evensen (1994), Burgers et al. (1998), Evensen (2007) and references therein.

Introduce the following notation, with

$$\begin{aligned} \mathbf{x}_t^u &= [\mathbf{x}_t|\mathbf{d}_0,...,\mathbf{d}_{t-1}] \\ \mathbf{x}_t^c &= [\mathbf{x}_t|\mathbf{d}_0,...,\mathbf{d}_t], \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots (6) \end{aligned}$$

where indices u and c indicate unconditioned and conditioned on the observation at the current time, respectively. Define a time series of ensembles:

$$\mathbf{e}_t : \{(\mathbf{x}_t^u,\mathbf{d}_t)^{(i)}; i = 1,...,n_{\mathbf{e}}\}; t = 0,...,T+1, \quad \ldots\ldots (7)$$

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

$$\mu_{\mathbf{xd}} = \begin{bmatrix} \mathrm{E}\{\mathbf{x}^u\} \\ \mathrm{E}\{\mathbf{d}\} \end{bmatrix} = \begin{bmatrix} \mu_{\mathbf{x}} \\ \mu_{\mathbf{d}} \end{bmatrix} \quad \ldots\ldots\ldots\ldots\ldots\ldots (8)$$

and

$$\Sigma_{\mathbf{xd}} = \begin{bmatrix} \mathrm{Cov}\{\mathbf{x}^u\} & \mathrm{Cov}\{\mathbf{x}^u,\mathbf{d}\} \\ \mathrm{Cov}\{\mathbf{d},\mathbf{x}^u\} & \mathrm{Cov}\{\mathbf{d}\} \end{bmatrix} = \begin{bmatrix} \Sigma_{\mathbf{x}} & \Gamma_{\mathbf{x},\mathbf{d}} \\ \Gamma_{\mathbf{d},\mathbf{x}} & \Sigma_{\mathbf{d}} \end{bmatrix}. \quad (9)$$

The traditional EnKF, see Evensen (2007), is defined with a Gauss-linear likelihood model

$$\mathbf{d}_t|\mathbf{x}_t = H_t\mathbf{x}_t + \varepsilon_t, \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots (10)$$

with ε_t being $N_{p_{\mathbf{x}}}(\mathbf{0},\Sigma_{\mathbf{d}|\mathbf{x}})$, or a linearization of a nonlinear likelihood model. The associated traditional EnKF algorithm is presented in Algorithm 1.

The EnKF algorithm 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.

Algorithm 1: Traditional Ensemble Kalman filter

Initiate:
- $n_{\mathbf{e}} =$ no. of ensemble members
- $\mathbf{x}_0^{u(i)}; i = 1,...,n_{\mathbf{e}}$ iid $f(\mathbf{x}_0)$
- $\varepsilon_0^{\mathbf{d}(i)} \sim N_{p_{\mathbf{d}}}(\mathbf{0},\Sigma_{\mathbf{d}|\mathbf{x}}); i = 1,...,n_{\mathbf{e}}$
- $\mathbf{d}_0^{(i)} = H_0\mathbf{x}_0^{u(i)} + \varepsilon_0^{\mathbf{d}(i)}; i = 1,...,n_{\mathbf{e}}$
- $\mathbf{e}_0 : \{(\mathbf{x}_0^u,\mathbf{d}_0)^{(i)}; i = 1,...,n_{\mathbf{e}}\}$

for $t = 0$ *to* T **do**

Conditioning:
- Estimate $\widehat{\Sigma}_{\mathbf{x}}$ from $\mathbf{e}_t$
- $\widehat{\Gamma}_{\mathbf{x},\mathbf{d}} = \widehat{\Sigma}_{\mathbf{x}}H_t'$
- $\widehat{\Sigma}_{\mathbf{d}} = H_t\widehat{\Sigma}_{\mathbf{x}}H_t' + \Sigma_{\mathbf{d}|\mathbf{x}}$
- $\mathbf{x}_t^{c(i)} = \mathbf{x}_t^{u(i)} + \widehat{\Gamma}_{\mathbf{x},\mathbf{d}}\widehat{\Sigma}_{\mathbf{d}}^{-1}(\mathbf{d}_t - \mathbf{d}_t^{(i)}); i = 1,...,n_{\mathbf{e}}$

Forwarding:
- $\varepsilon_t^{\mathbf{x}(i)} \sim N_{p_{\mathbf{x}}}(\mathbf{0},\mathbf{I}_{p_{\mathbf{x}}}); i = 1,...,n_{\mathbf{e}}$
- $\mathbf{x}_{t+1}^{u(i)} = \omega_t(\mathbf{x}_t^{c(i)},\varepsilon_t^{\mathbf{x}(i)}); i = 1,...,n_{\mathbf{e}}$
- $\varepsilon_{t+1}^{\mathbf{d}(i)} \sim N_{p_{\mathbf{d}}}(\mathbf{0},\Sigma_{\mathbf{d}|\mathbf{x}}); i = 1,...,n_{\mathbf{e}}$
- $\mathbf{d}_{t+1}^{(i)} = H_{t+1}\mathbf{x}_{t+1}^{u(i)} + \varepsilon_{t+1}^{\mathbf{d}(i)}; i = 1,...,n_{\mathbf{e}}$
- $\mathbf{e}_{t+1} : \{(\mathbf{x}_{t+1}^u,\mathbf{d}_{t+1})^{(i)}; i = 1,...,n_{\mathbf{e}}\}$

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

The forwarding operation is defined by the forward pdf. This entails two implicit approximations in the EnKF:

The sample space of $\mathbf{x}_t$ is discretized and represented by a finite number of realizations. Initially an 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.

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. Other problems arise in the EnKF which are caused by the use of an estimate of the Kalman gain based on $\mathbf{e}_t$ instead of the true weights. These problems include rank deficiency and estimation uncertainty due to the limited size of the ensemble, i.e., small values of $n_{\mathbf{e}}$. A discussion of the implications of data conditioning based on finite sample ensemble statistics follows.

The conditioning step. The conditioning step in the EnKF contains the linear approximation that appears crucial for the success of the filter. The conditioning expression relies on the Kalman gain $K = \Gamma_{\mathbf{x},\mathbf{d}}\Sigma_{\mathbf{d}}^{-1}$, which must be estimated at each state from the $n_{\mathbf{e}}$ members of the ensemble $\mathbf{e}_t$. In the general case with nonlinear likelihood model, the classical covariance estimators are applied:

$$\widehat{\Gamma}_{\mathbf{x},\mathbf{d}} = \frac{1}{n_{\mathbf{e}}-2}\sum_{i=1}^{n_{\mathbf{e}}}(\mathbf{x}_t^{u(i)} - \widehat{\mu}_{\mathbf{x}})(\mathbf{d}_t^{(i)} - \widehat{\mu}_{\mathbf{d}})' \quad \ldots\ldots\ldots\ldots (11)$$

$$\widehat{\Sigma}_{\mathbf{d}} = \frac{1}{n_{\mathbf{e}}-1}\sum_{i=1}^{n_{\mathbf{e}}}(\mathbf{d}_t^{(i)} - \widehat{\mu}_{\mathbf{d}})(\mathbf{d}_t^{(i)} - \widehat{\mu}_{\mathbf{d}})', \quad \ldots\ldots\ldots\ldots (12)$$

with

$$\widehat{\mu}_{\mathbf{x}} = \frac{1}{n_{\mathbf{e}}}\sum_{i=1}^{n_{\mathbf{e}}}\mathbf{x}_t^{u(i)} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots (13)$$

and

$$\widehat{\mu}_{\mathbf{d}} = \frac{1}{n_{\mathbf{e}}} \sum_{i=1}^{n_{\mathbf{e}}} \mathbf{d}_t^{(i)}. \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (14)$$

If $\mathbf{e}_t$ contains independent members these estimators are unbiased and consistent. The latter entails $\widehat{\Gamma}_{\mathbf{x},\mathbf{d}} \rightarrow \Gamma_{\mathbf{x},\mathbf{d}}$ and $\widehat{\Sigma}_{\mathbf{d}} \rightarrow \Sigma_{\mathbf{d}}$ as $n_{\mathbf{e}} \rightarrow \infty$, for all distributional models. Moreover, $\widehat{K} = \widehat{\Gamma}_{\mathbf{x},\mathbf{d}}\widehat{\Sigma}_{\mathbf{d}}^{-1} \rightarrow K$ as $n_{\mathbf{e}} \rightarrow \infty$, but note that for finite $n_{\mathbf{e}}$, $\widehat{K}$ is not unbiased due to non-linear dependence on the estimated covariance matrices. The consequences of this bias are thoroughly discussed in Furrer and Bengtsson (2007), where conditions on the size of $n_{\mathbf{e}}$ for obtaining bounded error growth is developed. It was recommended to use a boosting or inflation factor to correct the variability for this bias. For finite $n_{\mathbf{e}}$, it is known (Huber, 1981) that the classical estimators for covariance matrices are notoriously unreliable due to extreme dependence on the tail behavior of the underlying pdf. This sensitivity is caused by the second order terms of the estimators. The lack of precision in $\widehat{K}$ may cause spurious values to appear in the conditioned $\mathbf{x}_t^c$, which impact may be accelerated by non-linear forward models.

The motivation for this study follows from a closer evaluation of the conditioning relation

$$\mathbf{x}^{c(i)} = \mathbf{x}^{u(i)} + K(\mathbf{d} - \mathbf{d}^{(i)}); i = 1, ..., n_{\mathbf{e}}, \quad \ldots\ldots\ldots\ldots \quad (15)$$

where the time index is omitted for simplicity. Let the prior model for $\mathbf{x}^u$ be a Gaussian and the likelihood be Gauss-linear with known model parameters. Then the Kalman gain $K = \Sigma_{\mathbf{x}}H'(H\Sigma_{\mathbf{x}}H' + \Sigma_{\mathbf{d}|\mathbf{x}})^{-1}$ is known. For this case $\mathbf{x}^c$ is Gaussian with $\mathrm{E}\{\mathbf{x}^c\} = \mathrm{E}\{\mathbf{x}^u|\mathbf{d}\} = \mu_{\mathbf{x}|\mathbf{d}}$ and $\mathrm{Cov}\{\mathbf{x}^c\} = \mathrm{Cov}\{\mathbf{x}^u|\mathbf{d}\} = \Sigma_{\mathbf{x}|\mathbf{d}}$ which constitutes the correct solution. Moreover, if the ensemble members $\mathbf{x}_t^{u(i)}; i = 1, ..., n_{\mathbf{e}}$ are independent, the resulting $\mathbf{x}_t^{c(i)}; i = 1, ..., n_{\mathbf{e}}$ will also be independent.

In practice, however, the model parameters are not known and K must be estimated from the $n_{\mathbf{e}}$ members of the ensemble $\mathbf{e}$. Assume that the elements in $\mathbf{e} : \{(\mathbf{x}^u, \mathbf{d})^{(i)}; i = 1, ..., n_{\mathbf{e}}\}$ are independent. Hence K can be seen as a random variable $K_{\mathbf{e}} \sim f(K_{\mathbf{e}})$. In order to evaluate the characteristics of $\mathbf{x}^c$ when $K_{\mathbf{e}}$ is random, consider the Taylor expansion of Expression (15) around $\mathrm{E}\{K_{\mathbf{e}}\} = \mu_{K_{\mathbf{e}}}$ and $\mathrm{E}\{\mathbf{d}^{(i)}\} = \mu_{\mathbf{d}}$:

$$\begin{aligned}\mathbf{x}^{c(i)} \approx\ & \mathbf{x}^{u(i)} + \mu_{K_{\mathbf{e}}}(\mathbf{d} - \mu_{\mathbf{d}}) - \mu_{K_{\mathbf{e}}}(\mathbf{d}^{(i)} - \mu_{\mathbf{d}}) \\ & + (K_{\mathbf{e}} - \mu_{K_{\mathbf{e}}})(\mathbf{d} - \mu_{\mathbf{d}}) - \frac{1}{2}(K_{\mathbf{e}} - \mu_{K_{\mathbf{e}}})(\mathbf{d}^{(i)} - \mu_{\mathbf{d}}) \\ & ; i = 1, ..., n_{\mathbf{e}} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (16)\end{aligned}$$

Assume further that $(\mathbf{x}^u, \mathbf{d})^{(i)}$ and $K_{\mathbf{e}}$ are independent for all i, which is not unreasonable since $K_{\mathbf{e}}$ is based on all ensemble members.

Expression (16) provides $\mathrm{E}\{\mathbf{x}^c\} = \mu_{\mathbf{x}} + \mu_{K_{\mathbf{e}}}(\mathbf{d} - \mu_{\mathbf{d}})$ as an approximation to the conditional expectation $\mu_{\mathbf{x}|\mathbf{d}}$ and $\mathrm{Cov}\{\mathbf{x}^c\} = \Sigma_{\mathbf{x}} + \mu_{K_{\mathbf{e}}}\Sigma_{\mathbf{d}}\mu'_{K_{\mathbf{e}}} - 2\mu_{K_{\mathbf{e}}}\Gamma_{\mathbf{d},\mathbf{x}} + C(K_{\mathbf{e}})$, where $C(K_{\mathbf{e}})$ is a variance term representing uncertainty in the Kalman gain $K_{\mathbf{e}}$ as an approximation to the conditional covariance $\Sigma_{\mathbf{x}|\mathbf{d}}$. Note however that the cross ensemble covariance is $\mathrm{Cov}\{\mathbf{x}^{c(i)}, \mathbf{x}^{c(j)}\} = C(K_{\mathbf{e}})$, caused by $(\mathbf{x}^u, \mathbf{d})^{(i)}$ and $(\mathbf{x}^u, \mathbf{d})^{(j)}$ being independent while the same $K_{\mathbf{e}}$ is used for all ensemble members. Hence the members of the conditioned ensemble will be positively coupled and the empirical covariance matrix based on the ensemble will be underestimated. This is alarming since the EnKF is based on a sequential conditioning through time meaning that the coupling will grow increasingly stronger.

An alternative view on this dependence effect follows from considering the actual Kalman gain estimate $\widehat{K}_{\mathbf{e}}$ as given, i.e., using the plug-in approach. This entails that $\mathrm{E}\{\mathbf{x}^c|\widehat{K}_{\mathbf{e}}\} = \mu_{\mathbf{x}} + \mu_{K_{\mathbf{e}}}(\mathbf{d} - \mu_{\mathbf{d}}) + (\widehat{K}_{\mathbf{e}} - \mu_{K_{\mathbf{e}}})(\mathbf{d} - \mu_{\mathbf{d}})$ and $\mathrm{Cov}\{\mathbf{x}^c|\widehat{K}_{\mathbf{e}}\} = \Sigma_{\mathbf{x}} + \mu_{K_{\mathbf{e}}}\Sigma_{\mathbf{d}}\mu'_{K_{\mathbf{e}}} - 2\mu_{K_{\mathbf{e}}}\Gamma_{\mathbf{d},\mathbf{x}}$ while $\mathrm{Cov}\{\mathbf{x}^{c(i)}, \mathbf{x}^{c(j)}|\widehat{K}_{\mathbf{e}}\} = 0$. Hence for one particular EnKF run, the ensemble average will be biased and the empirical covariance matrix will be underestimated, not accounting for the Kalman gain estimation uncertainty. Note, however, that the Mean Squared Error (MSE), $\mathrm{E}\{(\mathbf{x}^c - (\mu_{\mathbf{x}} + \mu_{K_{\mathbf{e}}}(\mathbf{d} - \mu_{\mathbf{d}})))^2\}$ will capture the correct uncertainty, but the bias term will only be assessable through several EnKF runs. Lastly, note that as $n_{\mathbf{e}} \rightarrow \infty$ the uncertainty in $K_{\mathbf{e}}$ decreases and all problems disappear.

One possible solution to avoid this coupling problem is to perform Kalman gain resampling:

$$\begin{aligned}& K_{\mathbf{e}}^{(1)}, ..., K_{\mathbf{e}}^{(n_{\mathbf{e}})} \text{ iid } f, ..., (K_{\mathbf{e}}) \\ & \mathbf{x}^{c(i)} = \mathbf{x}^{u(i)} + K_{\mathbf{e}}^{(i)}(\mathbf{d} - \mathbf{d}^{(i)}); i = 1, ..., n_{\mathbf{e}} \quad \ldots\ldots\ldots \quad (17)\end{aligned}$$

Then $\mathrm{E}\{\mathbf{x}^c\}$ and $\mathrm{Cov}\{\mathbf{x}^c\}$ will remain the same as for Expression (15), while $\mathrm{Cov}\{\mathbf{x}^{c(i)}, \mathbf{x}^{c(j)}\} = 0$, and hence the ensemble coupling disappears. The assessment of $f(K_{\mathbf{e}})$ remains a challenge of course.

For the complete Gauss-linear model, all finite sample distributions are known. In particular it is known that $(n_{\mathbf{e}} - 1)\widehat{\Sigma}_{\mathbf{x}} \sim W(\Sigma_{\mathbf{x}}, n_{\mathbf{e}} - 1)$, i.e., Wishart distributed with parameters $(\Sigma_{\mathbf{x}}, n_{\mathbf{e}} - 1)$, see Mardia et al. (1979). The following resampling is reasonable:

$$\begin{aligned}& (n_{\mathbf{e}} - 1)\Sigma_{\mathbf{x}}^{(1)}, ..., (n_{\mathbf{e}} - 1)\Sigma_{\mathbf{x}}^{(n_{\mathbf{e}})} \text{ iid } W(\Sigma_{\mathbf{x}}, n_{\mathbf{e}} - 1) \\ & K_{\mathbf{e}}^{(i)} = \Sigma_{\mathbf{x}}^{(i)}H'(H\Sigma_{\mathbf{x}}^{(i)}H' + \Sigma_{\mathbf{d}|\mathbf{x}})^{-1}; i = 1, ..., n_{\mathbf{e}} \\ & \mathbf{x}^{c(i)} = \mathbf{x}^{u(i)} + K_{\mathbf{e}}^{(i)}(\mathbf{d} - \mathbf{d}^{(i)}); i = 1, ..., n_{\mathbf{e}} \quad \ldots\ldots\ldots \quad (18)\end{aligned}$$

Note that $\bar{K}_{\mathbf{e}} = \frac{1}{n_{\mathbf{e}}}\sum_{i=1}^{n_{\mathbf{e}}} K_{\mathbf{e}}^{(i)}$ will be an unbiased estimator for the Kalman gain K and hence some of the problems discussed in Furrer and Bengtsson (2007) may be of less concern.

The resampling EnKF approach specified in Expression (18) is primarily aimed at restoring the variability in the conditioned ensemble, and hence provide reliable prediction intervals. The variability in the estimated Kalman gains $K_{\mathbf{e}}$ will remain and spurious values in $\mathbf{x}^c$ will still occur. The Hierarchical EnKF (HEnKF) approach, presented in Myrseth and Omre (2009) aims at combining a shrinkage estimator for $K_{\mathbf{e}}$ which reduce spurious values and the resampling approach defined above. The empirical study in Myrseth and Omre (2009) provides very encouraging results, but HEnKF requires additional modeling which can be difficult in large problems. In the current paper we present an empirical resampling approach which requires no additional modeling assumptions.

Resampling

Resampling or simulating from a known pdf is called Monte Carlo simulation, see Hammersley and Handscomb (1964). This is sometimes the most efficient way to determine the pdf of random variables that are nonlinear functions of random variables with known pdfs. Actually, this is exactly what is done in the EnKF. The bootstrap, formally introduced in Efron (1979), is a statistical method to assess parameter uncertainty. Here we will only give a short introduction to the bootstrap and Monte Carlo techniques, and refer the interested reader to Efron and Tibshirani (1993).

Consider a random variable with an associated cdf, $\mathbf{x} \sim F(\mathbf{x})$, and some interesting characteristic of the cdf, $\xi = h(F(\mathbf{x}))$. Examples of this characteristic are the expectation $\mathrm{E}\{\mathbf{x}\}$, covariance $\mathrm{Cov}\{\mathbf{x}\}$, quantiles $\mathrm{Prob}\{\mathbf{x} \leq \mathrm{c}\}$ for some arbitrary c etc. Assume that a set of realizations $\mathbf{x}_1, ..., \mathbf{x}_n$ iid $F(\mathbf{x})$ are available and define a finite sample estimator of ξ, $\hat{\xi}_n = h_n(\mathbf{x}_1, ..., \mathbf{x}_n)$, such that $h_n(\mathbf{x}_1, ..., \mathbf{x}_n) \overset{n\rightarrow\infty}{\longrightarrow} h(F(\mathbf{x}))$. The objective is to obtain the cdf of the finite sample estimator $\hat{\xi}_n$, $F_n(\xi)$. If $F(\mathbf{x})$ is fully known, then assessment of $F_n(\xi)$ can be done by Monte Carlo simulation as described in Algorithm 2.

The approximation depends on the number of Monte Carlo samples, and the finite sample pdf can be fully determined when the number of Monte Carlo samples tends to infinity, $\hat{F}_n(\xi) \overset{m\rightarrow\infty}{\longrightarrow} F_n(\xi)$.

If $F(\mathbf{x})$ is unknown, however, Monte Carlo assessment is unavailable and one has to rely on the bootstrap technique. The random sample $\mathbf{x}_1, ..., \mathbf{x}_n$ iid $F(\mathbf{x})$ is used to obtain an estimate of $F(\mathbf{x})$, termed $\hat{F}(\mathbf{x})$. When $F(\mathbf{x})$ is completely unspecified, the non para-

Algorithm 2: Monte Carlo simulation

Initiate:

- $m =$ no. of Monte Carlo replicates

for $i = 1$ *to* m **do**

Generate: $\mathbf{x}_1^*, ..., \mathbf{x}_n^*$ iid $F(\mathbf{x})$

$\hat{\xi}_n^{(i)} = h_n(\mathbf{x}_1^*, ..., \mathbf{x}_n^*)$

Estimate: $F_n(\xi)$ from $\hat{\xi}_n^{(1)}, ..., \hat{\xi}_n^{(m)} \rightarrow \hat{F}_n(\xi)$

Algorithm 3: Bootstrap

Initiate:

- $b =$ no. of bootstrap replicates
- $\hat{F}(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} I(\mathbf{x}_i < \mathbf{x})$

for $i = 1$ *to* b **do**

Generate: $\mathbf{x}_1^*, ..., \mathbf{x}_n^*$ iid $\hat{F}(\mathbf{x})$

$\hat{\xi}_n^{*(i)} = h_n(\mathbf{x}_1^*, ..., \mathbf{x}_n^*)$

Estimate: $F_n(\xi)$ from $\hat{\xi}_n^{*(1)}, ..., \hat{\xi}_n^{*(b)} \rightarrow \hat{F}_n^*(\xi)$

metric estimate $\hat{F}(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} I(\mathbf{x}_i < \mathbf{x})$ can be used. The non-parametric bootstrap algorithm is given in Algorithm 3, see Efron (1979).

One important difference between bootstrapping and Monte Carlo simulation is that the bootstrap estimate of the finite sample pdf is not asymptotically correct when the number of bootstrap samples tends to infinity.

If the theoretical calculations of the fitted model are too complex, and the sample size is too small to have trustworthy approximations, parametric assumptions regarding the model can be imposed, see Davidson and Hinkley (1997). This is known as parametric bootstrapping. The non-parametric bootstrap imposes no additional assumptions on the model and is thus preferred.

Resampling the EnKF

The EnKF resampling is performed along the lines of Expression (17), recognizing that $K = \Gamma_{\mathbf{x},\mathbf{d}}\Sigma_{\mathbf{d}}^{-1}$. Hence the cdf of relevance is $F(\mathbf{x}^u, \mathbf{d}) = F(\mathbf{d}|\mathbf{x}^u)F(\mathbf{x}^u)$ from which the characteristics $\Gamma_{\mathbf{x},\mathbf{d}}$ and $\Sigma_{\mathbf{d}}$ can be determined. The associated finite sample estimators are the classical estimators given in Expression (12). The cdf $F(\mathbf{x})$ is only assessable through the ensemble members $(\mathbf{x}^{u(1)}, ..., \mathbf{x}^{u(n_\mathbf{e})})$, and may be bootstrapped by the nonparametric $\hat{F}(\mathbf{x}^u)$, see Algorithm 3. The conditional cdf $F(\mathbf{d}|\mathbf{x}^u)$ is defined by the likelihood model $\nu(\mathbf{x}^u, \varepsilon^\mathbf{d})$ which is fully specified by the known function $\nu(.,.)$ and the known pdf of $\varepsilon^\mathbf{d}$. Consequently, $F(\mathbf{d}|\mathbf{x}^u)$ can be assessed by Monte Carlo simulation, see Algorithm 2.

One resample replicate of the Kalman gain K^* in Expression (17) is generated by one bootstrap sample from $\hat{F}(\mathbf{x}^u)$, to obtain $(\mathbf{x}^{u*(1)}, ..., \mathbf{x}^{u*(n_\mathbf{e})})$. For each bootstrap sample, Monte Carlo sampling from $\hat{F}(\mathbf{d}|\mathbf{x}^{u*})$ is performed to obtain $[(\mathbf{x}^{u*(1)}, \mathbf{d}^{*(1,1)}), ..., (\mathbf{x}^{u*(n_\mathbf{e})}, \mathbf{d}^{*(n_\mathbf{e},1)}), ..., (\mathbf{x}^{u*(1)}, \mathbf{d}^{*(1,m)}), ..., (\mathbf{x}^{u*(n_\mathbf{e})}, \mathbf{d}^{*(n_\mathbf{e},m)})]$. Based on these realizations from $F(\mathbf{x}^u, \mathbf{d})$ the estimates $\widehat{\Gamma}_{\mathbf{x},\mathbf{d}}^*$ and $\widehat{\Sigma}_\mathbf{d}^*$ are computed to provide one resample replicate of the Kalman gain $K^* = \widehat{\Gamma}_{\mathbf{x},\mathbf{d}}^*(\widehat{\Sigma}_\mathbf{d}^*)^{-1}$.

For $n_\mathbf{e} >> \min(p_\mathbf{x}, p_\mathbf{d})$ full rank of $\widehat{\Gamma}_{\mathbf{x},\mathbf{d}}^*$ will be ensured. Note that if $n_\mathbf{e} < \min(p_\mathbf{x}, p_\mathbf{d})$, the rank of $\widehat{\Gamma}_{\mathbf{x},\mathbf{d}}^*$ will vary dependent on the number of duplicates in the bootstrap sample. The number of Monte Carlo replicates for each bootstrap sample can be chosen freely, hence full rank of $\widehat{\Sigma}_\mathbf{d}^*$ can be ensured.

The number of bootstrap replicates should be identical to the number of Kalman gain replicates required in Expression (17). Hence one replicate for each member in the unconditioned ensemble $(\mathbf{x}^{u(1)}, ..., \mathbf{x}^{u(n_\mathbf{e})})$ in order to perform the conditioning. If more replicates are used some of the unconditioned ensemble members must be used more than once, and hence unwanted coupling in the conditional ensemble will be introduced. Couplings in the ensemble will complicate assessments of prediction intervals at a later stage.

Algorithm 4: Resampled Ensemble Kalman Filter

Initiate:

- $n_\mathbf{e} =$ no. of ensemble members
- $m =$ no. of Monte Carlo replicates
- $\mathbf{x}_0^{u(i)}; i = 1, ..., n_\mathbf{e}$ iid $f(\mathbf{x}_0)$
- $\varepsilon_0^{\mathbf{d}(i)} \sim N_{p_\mathbf{d}}(\mathbf{0}, \mathbf{I}_{p_\mathbf{d}}); i = 1, ..., n_\mathbf{e}$
- $\mathbf{d}_0^{(i)} = \nu_t(\mathbf{x}_0^{u(i)}, \varepsilon_0^{\mathbf{d}(i)}); i = 1, ..., n_\mathbf{e}$
- $\mathbf{e}_0 : \{(\mathbf{x}_0^u, \mathbf{d}_0)^{(i)}; i = 1, ..., n_\mathbf{e}\}$

for $t = 0$ *to* T **do**

Conditioning:

Estimate $\hat{F}(\mathbf{x}_t^u) = \frac{1}{n}\sum_{i=1}^{n} I(\mathbf{x}_t^{u(i)} < \mathbf{x}_t^u)$

for $j = 1$ *to* $n_\mathbf{e}$ **do**

- $\mathbf{x}_t^{u(i)*} \sim \hat{F}(\mathbf{x}); i = 1, ..., n_\mathbf{e}$
- $\mathbf{G} = \mathbf{0}$
- $\mathbf{S} = \mathbf{0}$

for $k = 1$ *to* m **do**

- $\varepsilon_t^{\mathbf{d}(i),k} \sim N_{p_\mathbf{d}}(\mathbf{0}, \mathbf{I}_{p_\mathbf{d}}); i = 1, ..., n_\mathbf{e}$
- $\mathbf{d}_t^{(i),k} = \nu_t(\mathbf{x}_t^{u(i)*}, \varepsilon_t^{\mathbf{d}(i),k}); i = 1, ..., n_\mathbf{e}$
- $\mathbf{G} = \mathbf{G} + \frac{1}{n_\mathbf{e}-2}\sum_{i=1}^{n_\mathbf{e}}(\mathbf{x}_t^{u(i)*} - \widehat{\mu}_\mathbf{x})(\mathbf{d}_t^{(i),k} - \widehat{\mu}_\mathbf{d}^k)'$
- $\mathbf{S} = \mathbf{S} + \frac{1}{n_\mathbf{e}-1}\sum_{i=1}^{n_\mathbf{e}}(\mathbf{d}_t^{(i),k} - \widehat{\mu}_\mathbf{d}^k)(\mathbf{d}_t^{(i),k} - \widehat{\mu}_\mathbf{d}^k)'$

- $\widehat{\Gamma}_{\mathbf{x},\mathbf{d}}^* = \frac{1}{m}\mathbf{G}$
- $\widehat{\Sigma}_\mathbf{d}^* = \frac{1}{m}\mathbf{S}$
- $\mathbf{x}_t^{c(i)} = \mathbf{x}_t^{u(i)} + \widehat{\Gamma}_{\mathbf{x},\mathbf{d}}^*(\widehat{\Sigma}_\mathbf{d}^*)^{-1}(\mathbf{d}_t - \mathbf{d}_t^{(i)})$

Forwarding:

- $\varepsilon_t^{\mathbf{x}(i)} \sim N_{p_\mathbf{x}}(\mathbf{0}, \mathbf{I}_{p_\mathbf{x}}); i = 1, ..., n_\mathbf{e}$
- $\mathbf{x}_{t+1}^{u(i)} = \omega_t(\mathbf{x}_t^{c(i)}, \varepsilon_t^{\mathbf{x}(i)}); i = 1, ..., n_\mathbf{e}$
- $\varepsilon_{t+1}^{\mathbf{d}(i)} \sim N_{p_\mathbf{d}}(\mathbf{0}, \mathbf{I}_{p_\mathbf{d}}); i = 1, ..., n_\mathbf{e}$
- $\mathbf{d}_{t+1}^{(i)} = \nu_{t+1}(\mathbf{x}_{t+1}^{u(i)}, \varepsilon_{t+1}^{\mathbf{d}(i)}); i = 1, ..., n_\mathbf{e}$
- $\mathbf{e}_{t+1} : \{(\mathbf{x}_{t+1}^u, \mathbf{d}_{t+1})^{(i)}; i = 1, ..., n_\mathbf{e}\}$

Assess

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

The additional computational demands from the resampling strategy are to recompute the Kalman gain $n_\mathbf{e}$ times in the bootstrapping and to recompute the likelihood function m times in the Monte Carlo step. The former will normally be relatively inexpensive, while the cost of the latter depends on the actual model.

The procedure described above defines the Resample EnKF algorithm, coined ResEnKF, see Algorithm 4. The basis for ResEnKF is only the ensemble members and the given likelihood model. No additional model assumptions are made. The resampled Kalman gains $K_\mathbf{e}^{(1)}, ..., K_\mathbf{e}^{(n_\mathbf{e})}$ will not be independent due to coupling through the ensemble. They will, however, reproduce more of the variability than the single Kalman gain estimate $\widehat{K}$ will. Consequently, the coupling will be reduced. The resulting ensemble improvements in the forecast at time $T+1$ will be evaluated in the following section.

Empirical study

In order to evaluate the impact of ResEnKF, we define a Gaussian prior model and two different likelihood models, one Gauss-linear and one non-linear. The Gauss-linear model with all model parameters known is analytically tractable and will act as a reference. This model is described in Myrseth and Omre (2009). The model with a nonlinear observation likelihood demonstrates the generality of the

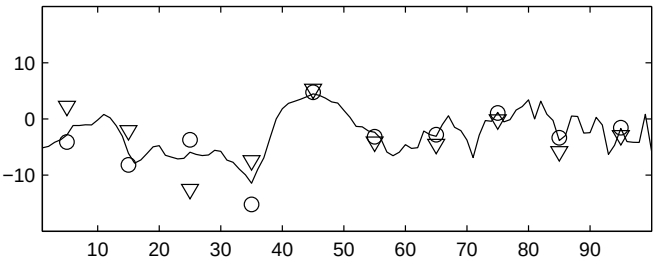

Fig. 2—Reference realization: $(\mathbf{x}_{10}, \mathbf{d}_{10})$. Observations from the linear likelihood marked by triangles. Observations from the nonlinear likelihood marked by circles.

	$n_\mathbf{e}$	RMSE	Coverage (%)
Exact solution		2.68	95.0
Exact finite sample solution	30	2.75	97.3
Traditional EnKF	30	3.55	62.3
ResEnKF	30	3.92	74.0
Exact finite sample solution	100	2.70	98.1
Traditional EnKF	100	2.93	88.8
ResEnKF	100	3.00	93.5

Table 1—RMSE and coverage for the different algorithms with a Gauss-linear likelihood model averaged over 100 runs with different initial seed.

algorithm.

Model description. The variables of interest are $[\mathbf{x}_0, ..., \mathbf{x}_{11}]$, 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 2 the reference realization of $[\mathbf{x}_{10}, \mathbf{d}_{10}]$ is presented.

The test case is defined as follows:

$$f(\mathbf{x}_0) \sim N_{100}(\mathbf{0}, \Sigma_0^x) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (19)$$

$$[\mathbf{x}_{t+1}|\mathbf{x}_t] = \mathbf{A}_t \mathbf{x}_t \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (20)$$

where the initial covariance matrix Σ_0^x contains elements

$$\sigma^x_{i,j} = 20 \exp(-3|i-j|/20) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (21)$$

for $i, j = 1, ..., 100$. The forward model defined by $\mathbf{A}_t$ is a linear smoother 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 example has been inspired by a fluid flow scenario where there is a moving front where the parameters are dynamic, and static surroundings. For more detail, see Myrseth and Omre (2009).

The likelihood models are

$$[\mathbf{d}_t|\mathbf{x}_t]_0 = H_t \mathbf{x}_t + \sqrt{20}\varepsilon_t^\mathbf{d} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (22)$$

and

$$[\mathbf{d}_t|\mathbf{x}_t]_1 = (H_t \mathbf{x}_t) \circ \exp(\sqrt{0.1}\varepsilon_t^\mathbf{d}) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (23)$$

where $\varepsilon_t^\mathbf{d} \sim N_{10}(\mathbf{0}, \mathbf{I}_{10})$, H_t is time-invariant and picks 10 locations, see Figure 2 and $\circ$ denotes a Schur product. The nonlinear likelihood contains a log normal multiplicative error structure.

Results. The forecast $[\mathbf{x}_{11}|\mathbf{d}_0, ..., \mathbf{d}_{10}]$ will be used to measure the impact of resampling. The Root Mean Squared Error (RMSE) of the mean forecast will be used to measure accuracy. The coverage will be used to measure forecast uncertainty which captures both accuracy and precision. A 95% coverage interval should include the solution 95% of the time. If the coverage is lower than this then the 95% forecast interval underestimates the uncertainty. The examples are run with the ensemble sizes $n_\mathbf{e} = 30$ and $n_\mathbf{e} = 100$. An empirical 95% prediction interval is defined to be spanned by the 28 and 96 central ensemble members for the two ensemble sizes. We should therefore expect 93.3% and 96% coverage respectively.

Gauss-linear likelihood. For the Gauss-linear model, the model parameters are available through the traditional Kalman Filter. For the Gauss-linear model also the resampling approach using the Wishart pdf as outlined in Expression (18) is available. This case is termed the exact finite sample solution, and it captures the uncertainty due to finite size of the ensemble. The traditional EnKF algorithm, Algorithm 1, and the ResEnKF, Algorithm 4, adapted to a known linear likelihood are also run on this Gauss-linear case.

Figure 3 and Table 1 display the results obtained for the Gauss-linear model. In Figure 3 the prediction $[\mathbf{x}_{11}, \mathbf{d}_1, ..., \mathbf{d}_{10}]$ with associated 95% prediction intervals are displayed for one run of each of the algorithms. The reference $\mathbf{x}_{11}$ is also displayed. Table 1 contains statistics from 100 repeated runs of each algorithm on the same observations.

Figure 3a) contains the analytical solution of the Kalman Filter which is available for this Gauss-linear model. The reduction in prediction uncertainty around each observation is observed. Figures 3b) and 3c) present the prediction results for the exact finite sample solutions for $n_\mathbf{e} = 30$ and $n_\mathbf{e} = 100$. Recall that the Kalman gains are resampled using the Wishart sampling pdf which adds uncertainty relative to the analytical solution which is the limiting case as $n_\mathbf{e} \to \infty$. The $n_\mathbf{e} = 30$ case has larger uncertainty while the $n_\mathbf{e} = 100$ case is very similar to the limiting case. These exact finite sample solutions are the reference solutions for the other ensemble Kalman filter runs. Figures 3d) and 3e) contain the results for the traditional EnKF algorithm. The underestimation of the prediction intervals for $n_\mathbf{e} = 30$ is observed and the non-logical increase in uncertainty as $n_\mathbf{e}$ increases is observed for $n_\mathbf{e} = 100$. We interpret the underestimation of uncertainty for $n_\mathbf{e} = 30$ to be caused by coupling of ensemble members due to the use of on common estimate of the Kalman gain, as discussed in previous sections. Note that all estimated covariance matrices have full rank in this case. The fact that the exact finite sample solutions, which uses independent Kalman gains for each ensemble member, exposes decreasing uncertainty with increasing $n_\mathbf{e}$, supports our interpretation. Figures 3f) and 3g) contain the results from the ResEnKF algorithm which includes bootstrapping of the ensemble members to provide Kalman gain variability and reduce coupling in the conditional ensembles. The prediction intervals for ResEnKF are wider than for traditional EnKF and close to the reference exact finite sample solutions in Figures 3b) and 3c). This effect is very clear for $n_\mathbf{e} = 30$, while the results are very similar for all algorithms for $n_\mathbf{e} = 100$.

The results from the Gauss-linear model case can be summarized as follows: The traditional EnKF appears with better prediction accuracy than the ResEnKF algorithm, but the latter assesses the prediction uncertainty more reliably. In the traditional EnKF one uses the best estimate of the Kalman gain on all ensemble members to improve prediction accuracy, but this introduces coupling in the ensemble and hence underestimation of prediction uncertainty. In ResEnKF one resamples the Kalman gain causing loss in prediction accuracy , but this also reduces coupling in the ensemble and hence improves the prediction uncertainty estimates. Lastly, note that these results are obtained on a Gauss-linear model which appears as very favorable for the EnKF. Hence the underestimation of the prediction uncertainty should cause concern. The effect on non-Gauss-linear models is unclear of course.

Nonlinear likelihood. For the nonlinear likelihood the true model parameters are analytically intractable. The traditional EnKF can be used with a prior linearization of the likelihood model:

$$[\mathbf{d}_t|\mathbf{x}_t]_1 \approx H_t \mathbf{x}_t + H_t \mu_\mathbf{x} \circ (\sqrt{0.1}\varepsilon_t^\mathbf{d}). \quad \ldots\ldots\ldots\ldots\ldots \quad (24)$$

This Taylor expansion is centered at the prior mean $\mu_\mathbf{x}$, and defines an additive Gaussian error term with adjusted variance. This approach is termed traditional EnKF/linearized likelihood, Algorithm 1. Alternatively, the resampling EnKF algorithm, Algorithm 4,

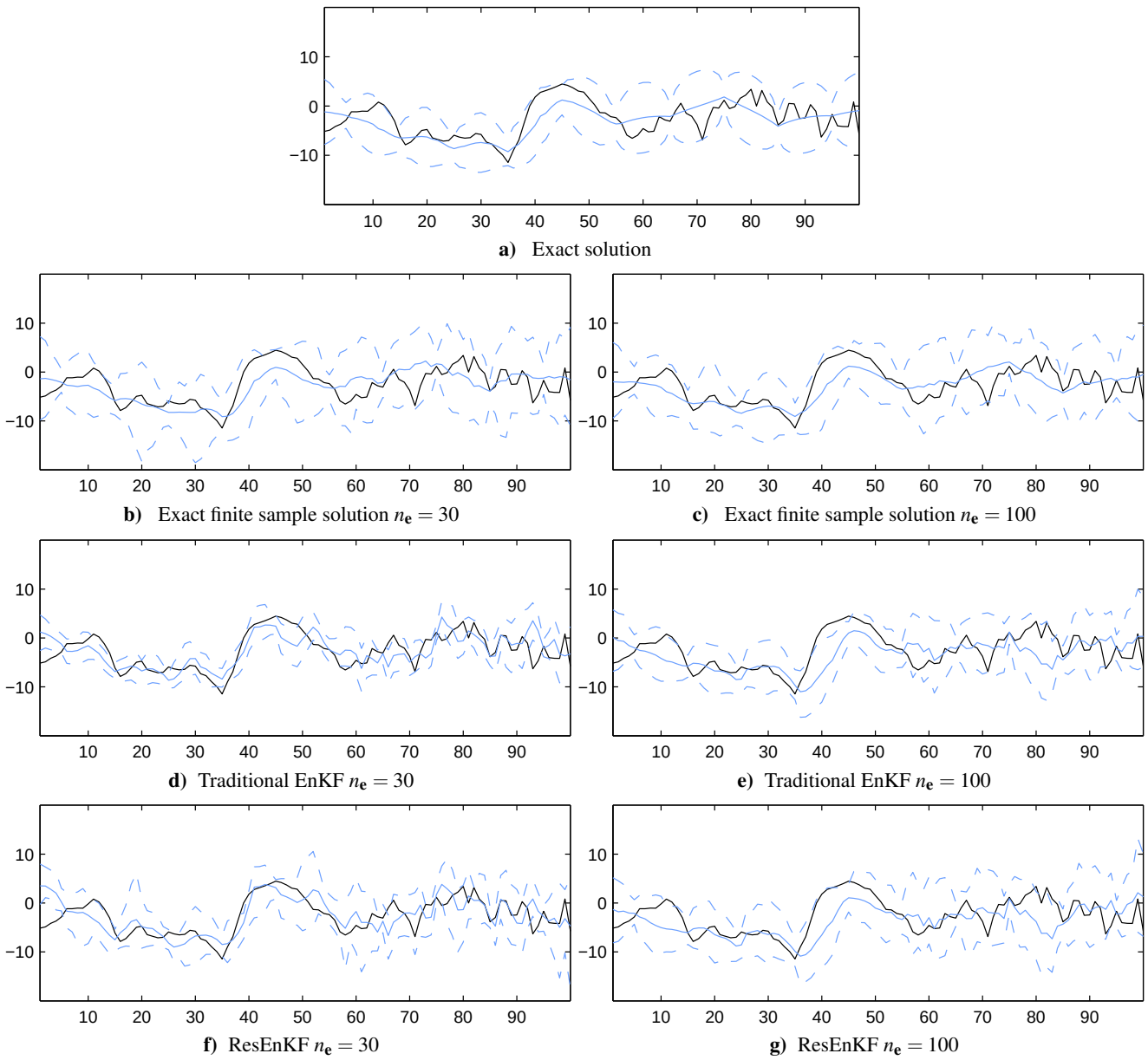

Fig. 3—The reference realization (black), predictions (solid blue) and 95%-empirical prediction intervals (hatched blue) for one run of the EnKF algorithm with different ensemble sizes.

	n_e	RMSE	Coverage (%)
Trad. EnKF / Linearized likelihood	30	5.56	39.8
EnKF / nonlinear likelihood	30	4.67	40.1
ResEnKF	30	5.81	67.4
Trad. EnKF / Linearized likelihood	100	3.84	74.8
EnKF / nonlinear likelihood	100	2.95	82.0
ResEnKF	100	3.10	93.0

Table 2—RMSE and coverage for the different algorithms with a nonlinear likelihood model averaged over 100 runs with different initial seed.

can be used without the bootstrapping loop. Then the likelihood model is linearized by Monte Carlo sampling based linearization around the ensemble. Hence, local ensemble dependent linearization is performed. This approach is termed EnKF/nonlinear likelihood. Finally, the full ResEnKF algorithm, Algorithm 4, including both bootstrapping and Monte Carlo sampling can be used.

Figure 4 and Table 2 display the results obtained for the Gaussian prior model with nonlinear likelihood model. The layout is identical to Figure 3 and Table 1.

Figures 4a) and 4b) contain the traditional EnKF solution based on a prior linearization of the likelihood model. Since the linearization is global some extreme observations appear to have too large influence. The prediction uncertainty is clearly underestimated for $n_e = 30$ and uncertainty increases with increasing n_e which is non-intuitive. Figures 4c) and 4d) display the EnKF solutions with local, ensemble based linearization of the likelihood model. The accuracy of the prediction appears better, but prediction uncertainties are underestimated similar to Figures 4a) and 4b). Figures 4e) and 4f) contain the results from the ResEnKF algorithm with bootstrap and Monte Carlo resampling. The prediction intervals for $n_e = 30$ appear as much more reliable than for the two other algorithms, while intervals for $n_e = 100$ are fairly similar.

In Table 2 statistic for 100 repeated runs of each algorithm are summarized. The traditional EnKF algorithm with prior, global lin-

earization has low prediction accuracy and underestimates the prediction uncertainty for both $n_{\mathbf{e}} = 30$ and $n_{\mathbf{e}} = 100$. The EnKF with local, ensemble based linearization has improved prediction accuracy, but tends to underestimate the prediction uncertainty. Lastly, the ResEnKF algorithm has higher RMSE than the EnKF with local linearization, but provides more reliable estimates of the prediction intervals.

The results from the Gaussian prior model with nonlinear likelihood model can be summarized as follows: An EnKF algorithm with local ensemble based Monte Carlo linearization should be used. By using the full resampling in the ResEnKF more reliable prediction uncertainty estimates are obtained, on the cost of somewhat lower prediction accuracy. This latter result is consistent with the experience from the Gauss-linear model.

Reservoir Example

In this section the traditional EnKF, the ResEnKF and the HEnKF-algorithms are implemented on a small realistically inspired synthetic reservoir. The reservoir example is described in detail in Myrseth and Omre (2009). The objective is to estimate the permeability field based on production data and time lapse seismic surveys. In this example the number of reservoir parameters, $n_{\mathbf{x}}$, and the number of observations at each time, $n_{\mathbf{d}}$, are both large. To handle this large model a localization scheme is used, but even then there are rank issues as $n_{\mathbf{e}} << \min(n_{\mathbf{x}}, n_{\mathbf{d}})$.

Reservoir and model description. The reservoir under study is of size 640m $\times$ 640m $\times$ 40m modeled as $16 \times 16 \times 10 = 2560$ grid blocks. There are four injector wells, one placed in each corner, and a producer well in node (8,8) seen from above, see Figure 5. The state variable is modeled as

$$\mathbf{x}_t = \begin{bmatrix} \phi_t \\ k_t \\ s_t \\ \\ m_t \end{bmatrix} = \begin{bmatrix} poro \\ \log perm \\ \text{logit}\, sat \\ \begin{cases} \log v_p \\ \log v_s \end{cases} \\ \log \rho \end{bmatrix} \quad \ldots\ldots\ldots\ldots\ldots \quad (25)$$

where *poro* is porosity, *perm* permeability, *sat* water saturation, v_p, v_s and ρ are p-wave velocity, s-wave velocity and density in each grid node. The dimension of the state variable is thus $6 \times 2560 = 15360$.

The forward model can be formulated as the following first order Markov model;

$$\begin{aligned}[\mathbf{x}_t|\mathbf{x}_{t-1}] &= \begin{bmatrix} k_t|k_{t-1} \\ s_t|\phi_{t-1}, k_{t-1}, s_{t-1} \\ m_t|\phi_{t-1}, k_{t-1}, s_{t-1} \end{bmatrix} \\ &= \begin{bmatrix} k_{t-1} \\ w(\phi_{t-1}, k_{t-1}, s_{t-1}) \\ h(\phi_{t-1}, w(\phi_{t-1}, k_{t-1}, s_{t-1})) \end{bmatrix}, t = 1, 2. \quad (26)\end{aligned}$$

where ϕ and k are static variables. Saturation is forwarded based on a reservoir flow simulator $w(.)$, and m_t is forwarded based on Gassmann's quasi-static equations, $h(.)$, as described in Bachrach (2006). The reservoir simulator uses a two point flux approximation for the pressure and streamlines for the saturation, see Sintef (2007) and updates s_t which in turn updates m_t. In this example there is no error associated with the forward models. The reservoir is initially fully saturated with oil and water is injected through the injector wells.

The mean saturation in the producing well and seismic data are gathered at times $t = 0, 1, 2$ representing production days 0, 500 and 1000. Synthetic seismic data are generated trace-wise via the linearized AVO forward model described in Buland and Omre (2003). That is, observations are modeled as $[d_t^m|x_t] = WADm_t + \varepsilon_t^m$, where d_t^m are seismic responses for angles 0, 10 and 20 degrees, W is a Ricker wavelet of length 10 grid blocks, A is a linearized version of the Zoepprits equations, D is a difference matrix and ε_t^m is a Gaussian white noise term.

	$n_{\mathbf{e}}$	RMSE	Coverage (%)
EnKF	40	85.5	15.5
ResEnKF	40	95.6	61.1
HEnKF	40	52.2	82.9

Table 3—RMSE and coverage for the traditional EnKF, the ResEnKF and the HEnKF.

For description of the reference 'truth' and conditioning data see Myrseth and Omre (2009). The traditional EnKF algorithm, Algorithm 1, is applicable on this case since the likelihood model is defined as linear. In the ResEnKF algorithm only the bootstrap resampling is active since the likelihood model is linear. The parametrization of the HEnKF algorithm is identical to the case described in Myrseth and Omre (2009) and can be found there.

Figure 6 and Table 3 display the results obtained from this reservoir example. In Figure 6 the predicted permeability with associated empirical 95% prediction intervals at time $t = 1000$ for layers 2,4,6 and 8 for one run of the three algorithms is presented. The 16×16-dimensional layers are plotted as 16 consecutive 16×1 vectors to facilitate inspection. The true permeability values are also displayed.

The results in Figure 6 show clearly that the prediction intervals from the traditional EnKF are underestimated. The ResEnKF algorithm provides wider prediction intervals that appear more reliable. The prediction intervals from the HEnKF algorithm are wider and more smooth than for the ResEnKF algorithm. From Table 3 it is seen that the RMSE and coverage of the HEnKF algorithm is best. The ResEnKF algorithm provides more reliable prediction intervals on the cost of prediction accuracy compared to the traditional EnKF.

The results from the reservoir example can be summarized as follows: The HEnKF algorithm is favorable, and this is not surprising since more model assumptions are made through priors on model parameters. By using the ResEnKF more reliable prediction intervals are obtained on the cost of prediction accuracy. These results are consistent with other results in our study.

Conclusion

The traditional EnKF is based on an ensemble representing the pdf of the variable of interest. The ensemble members are sequentially conditioned on observations and forwarded to the next time step. The conditioning to available observations is the challenging part, and in EnKF this conditioning is linearized using weights corresponding to the Kalman gain. The actual Kalman gain is estimated based on all ensemble members. Since the same Kalman gain estimate is used in all the conditioning of all ensemble members it can be shown that the members end up being coupled. Eventually this will cause the prediction intervals to be underestimated. A resampling strategy where the Kalman gain is generated from its sampling distribution is suggested. This will prevent coupling in the conditioning step. For a Gauss-linear model this sampling distribution can be analytically assessed, and exact finite sample prediction intervals can be generated by one resampling EnKF run.

The resampling EnKF, ResEnKF, algorithm is defined for a model with both prior and the likelihood being on general nonlinear form. The likelihood is linearized by a local ensemble based Monte Carlo resampling strategy. The coupling is reduced by bootstrapping the ensemble members. The computational demands of ResEnKF are larger than for EnKF, but only slightly larger.

The ResEnKF algorithm is evaluated empirically. A Gauss-linear model is used and the exact finite sample prediction intervals are generated as reference. It is shown that the traditional EnKF algorithm severely underestimates the prediction intervals for small ensemble sizes. The ResEnKF has significantly higher coverage of the prediction intervals on the expense of somewhat larger mean squared error of prediction itself when compared to the EnKF. It is surprising that the effects are so large even for this Gauss-linear case which appears as very favorable for the traditional EnKF.

The algorithms are also evaluated on a Gaussian prior model

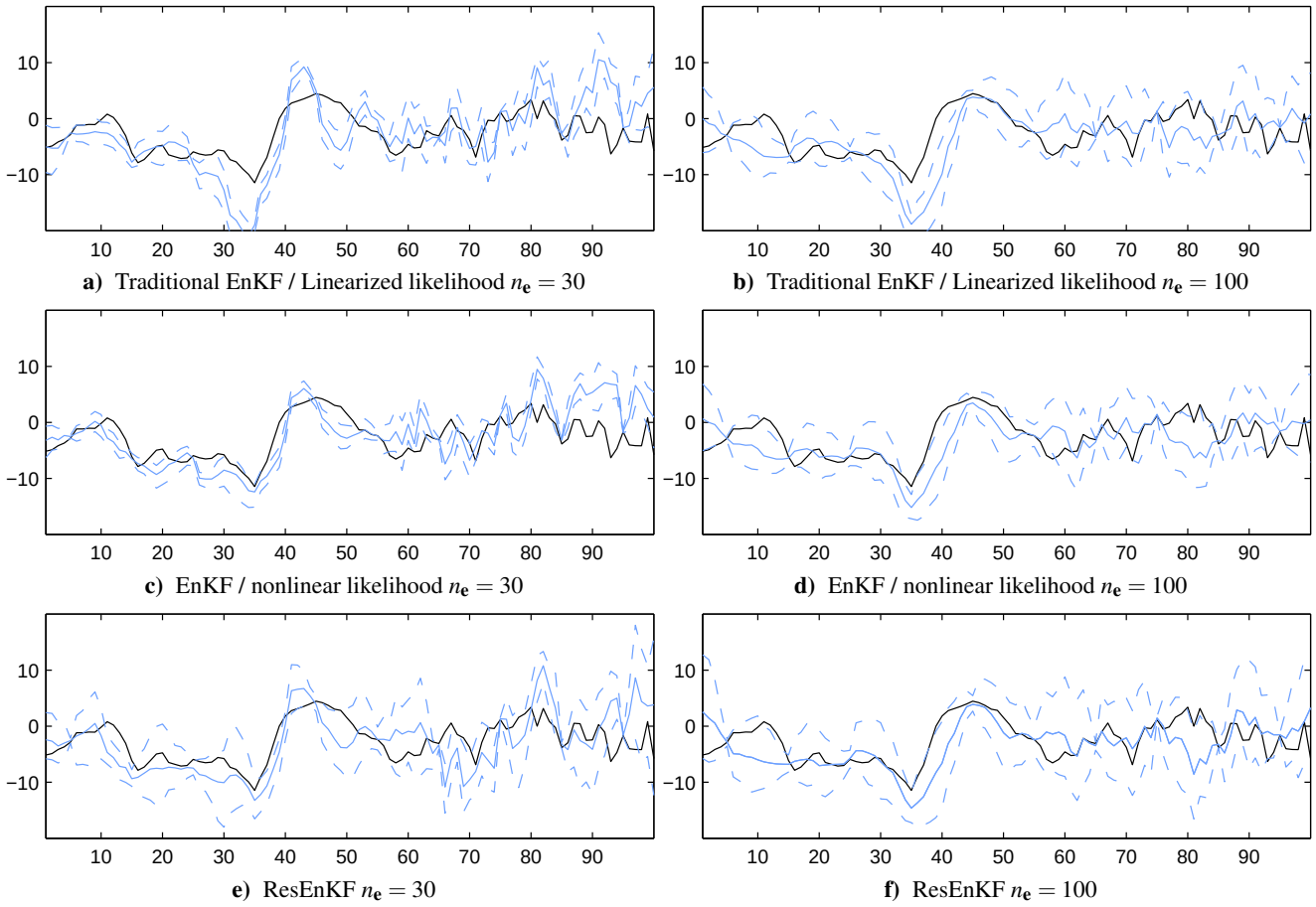

Fig. 4—The reference realization (black), predictions (solid blue) and 95%-empirical prediction intervals (hatched blue) for one run of the EnKF algorithm with different ensemble sizes.

with nonlinear likelihood model. The ResEnKF algorithm including Monte Carlo linearization and bootstrapping appeared as much more reliable than the traditional EnKF with prior linearization of the likelihood model.

Lastly, the EnKF, ResEnKF and HEnKF algorithms are compared on a synthetic reservoir production history conditioning case. The HEnKF provided the best results on the expense of considerably more modeling through priors on model parameters. The ResEnKF algorithm outperformed the traditional EnKF by providing much more reliable prediction intervals on the expense of slightly lower prediction accuracy. The ResEnKF algorithm requires no extra modeling and has only slightly larger computational demands than the EnKF algorithm.

The underestimation of prediction intervals from the EnFK algorithm is considerable for small ensemble sizes. By applying the ResEnKF algorithm more reliable prediction intervals can be provided with little additional computational demands.

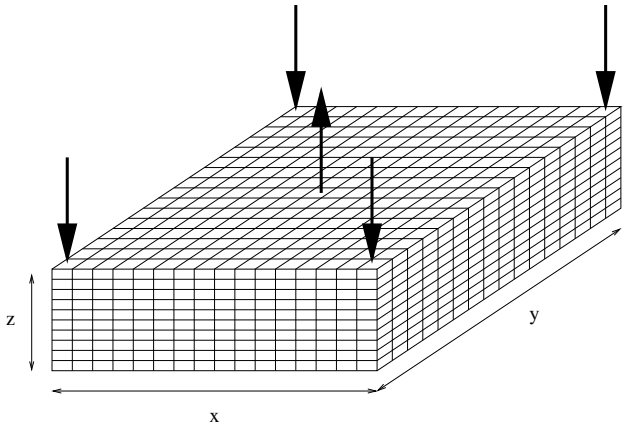

Fig. 5—A schematic figure of the synthetic reservoir. x, y and z are 640m, 640m and 40m divided into 16, 16 and 10 grid blocks respectively. The arrows indicate injector (downward) and producer (upward) wells.

Acknowledgments

This work is funded by the Uncertainty in Reservoir Evaluation (URE) initiative at NTNU. Thanks to Sintef for letting us use their reservoir simulator.

References

Anderson, J. 2001. An Ensemble Adjustment Ensemble Kalman Filter for Data Assimilation. *Monthly Weather Review*, **129**: 2884–2903. doi: 10.1175/1520-0493(2001)129<2884:AEAKFF>2.0.CO;2.

Bachrach, R. 2006. Joint Estimation of Porosity and Saturation using Stochastic Rock-Physics Modeling. *Geophysics*, **71**(5): O53–O63. doi: 10.1190/1.2235991.

Bertino, L., Evensen, G. and Wackernagel, H. 2003. Sequential Data Assimilation Techniques in Oceanography. *International Statistical Review*, **71**(2): 223–241. doi: 10.1111/j.1751-5823.2003.tb00194.x.

Buland, A. and Omre, H. 2003. Bayesian Linearized AVO Inver-

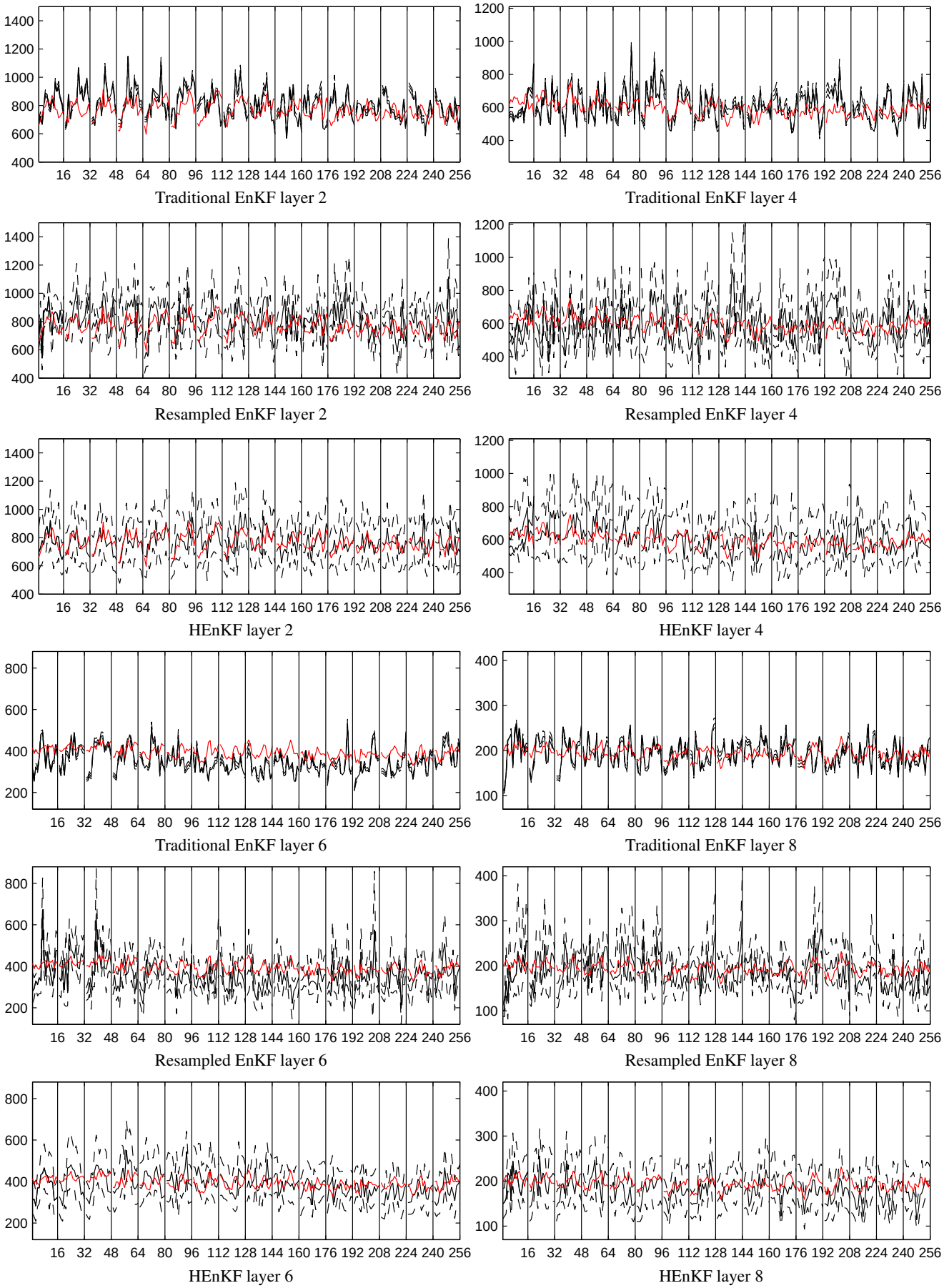

Fig. 6—The reference permeability (red), predictions (solid black) and 95%-empirical prediction intervals (hatched black) for one run of the EnKF and bootstrapped EnKF.

sion. *Geophysics*, **68**(1): 185–198. doi: 10.1190/1.1543206.

Burgers, G., van Leeuwen, P. J. and Evensen, G. 1998. Analysis Scheme in the Ensemble Kalman Filter. *Monthly Weather Review*, **126**(6): 1719–1724. doi: 10.1175/1520-0493(1998)126<1719:ASITEK>2.0.CO;2.

Davidson, A. C. and Hinkley, D. V. 1997. *Bootstrap Methods and their Application*. Cambridge University Press.

Efron, B. 1979. Bootstrap Methods: Another Look at the Jackknife. *The Annals of Statistics*, **7**(1): 1–26.

Efron, B. and Tibshirani, R. 1993. *An Introduction to the Bootstrap*. New York City: Chapman & Hall.

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. 2007. *Data Assimilation; The Ensemble Kalman Filter*. Berlin Heidelberg: Springer-Verlag.

Furrer, R. and Bengtsson, T. 2007. Estimation of High-dimensional Prior and Posteriori Covariance Matrices in Kalman Filter Variants. *Journal of Multivariate Analysis*, **98**(2): 227–255. doi: 10.1016/j.jmva.2006.08.003.

Hammersley, J. M. and Handscomb, D. C. 1964. *Monte Carlo Methods*. London & New York: Chapman & Hall.

Houtekamer, P., Mitchell, H. L., 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. doi: 10.1175/MWR-2864.1.

Huber, P. J. 1981. *Robust Statistics*. New York City: Wiley.

Mardia, K., Kent, J. and Bibby, J. 1979. *Multivariate Analysis*. London: Academic Press.

Myrseth, I. and Omre, H. 2009. Hierarchical Ensemble Kalman Filter. *SPE Journal*. Accepted for publication.

Nævdal, G., Johnsen, L. M., Aanonsen, S. I. and Vefring, E. H. 2005. Reservoir Monitoring and Continuous Model Updating Using Ensemble Kalman Filter. *SPE Journal*, **10**(1): 66–74. SPE-84372-PA. doi: 10.2118/84372-PA.

Sintef 2007. In-house reservoir flow simulator developed by Sintef under the GeoScale project. http://www.sintef.no/Projectweb/GeoScale/Projects/Geoscale/.