International Journal for Uncertainty Quantification, 4 (4): 349–364 (2014)

OPTIMIZATION-BASED SAMPLING IN ENSEMBLE KALMAN FILTERING

Antti Solonen,[1,3,*] Alexander Bibov,[1] Johnathan M. Bardsley,[2] & Heikki Haario[1,3]

[1] *Lappeenranta University of Technology, Laboratory of Applied Mathematics, Lappeenranta, Finland*

[2] *University of Montana, Department of Mathematical Sciences, Montana*

[3] *Finnish Meteorological Institute, Helsinki, Finland*

Original Manuscript Submitted: 04/18/2013; Final Draft Received: 01/27/2014

In the ensemble Kalman filter (EnKF), uncertainty in the state of a dynamical model is represented as samples of the state vector. The samples are propagated forward using the evolution model, and the forecast (prior) mean and covariance matrix are estimated from the ensemble. Data assimilation is carried out by using these estimates in the Kalman filter formulas. The prior is given in the subspace spanned by the propagated ensemble, the size of which is typically much smaller than the dimension of the state space. The rank-deficiency of these covariance matrices is problematic, and, for instance, unrealistic correlations often appear between spatially distant points, and different localization or covariance tapering methods are needed to make the approach feasible in practice. In this paper, we present a novel way to implement ensemble Kalman filtering using optimization-based sampling, in which the forecast error covariance has full rank and the need for localization is diminished. The method is based on the randomize then optimize (RTO) technique, where a sample from a Gaussian distribution is computed by perturbing the data and the prior, and solving a quadratic optimization problem. We test our method in two benchmark problems: the 40-dimensional Lorenz '96 model and the 1600-dimensional two-layer quasi-geostrophic model. Results show that the performance of the method is significantly better than that of the standard EnKF, especially with small ensemble sizes when the rank-deficiency problems in EnKF are emphasized.

KEY WORDS: *data assimilation, state estimation, ensemble Kalman filter, optimization-based sampling*

1. INTRODUCTION

In dynamical state estimation, or data assimilation, the distribution of the current state of a dynamical model is estimated by combining information from the forecast of the current state with new measurements that become available. Viewed from the Bayesian perspective, the forecast acts as the *prior*, which is combined with the *likelihood* to obtain the *posterior* distribution. If the forecast and observation models are linear, and errors are assumed to be normally distributed, the mean and the covariance matrix of the state can be computed with the Kalman filter (KF) formulas [1]. In the extended Kalman filter (EKF), the models are linearized and the KF formulas are applied. However, in high-dimensional systems, often encountered, e.g., in geophysics, the required matrix manipulations are prohibitively expensive. The current practice in large-scale data assimilation, such as weather prediction, relies mostly on variational methods, such as 3D-Var [2] and 4D-Var [3].

Recently, ensemble filtering methods that use Monte Carlo approximations for the covariance matrices have become popular. In ensemble filtering, initiated by the introduction of the ensemble Kalman filter (EnKF, [4]), the covariance matrices are essentially replaced with sample covariance matrices estimated from the ensemble. Posterior

*Correspond to Antti Solonen, E-mail: solonen@lut.fi, URL: http://solbes.piimakassi.fi/

2152–5080/14/$35.00

ensemble members are generated by applying the Kalman filter formulas to randomly perturbed observations. In the so-called square root ensemble filters [5–7], the prior ensemble is deterministically transformed into the posterior ensemble, and perturbed observations are not used.

Ensemble filters suffer from a few known problems. Since the dimension in geophysical applications can be very large and the forward model computationally heavy, the number of samples used in ensemble filtering is typically many orders of magnitude smaller than the dimension of the state space. This means that the prior covariance matrices are rank-deficient, and the analysis is restricted to the subspace spanned by the forecast ensemble, which can lead to ensemble inbreeding [8]. In addition, the prior covariance estimated from the samples can yield unrealistic, strong correlations between points that are, for instance, spatially distant [7, 9]. For this purpose, different localization and covariance tapering approaches have been developed, where the unrealistic correlations are cut off from the sample covariance matrix [10, 11]. Moreover, many ensemble methods underestimate the prior covariance matrices [12], which can lead to filter divergence, and different covariance inflation mechanisms have been proposed to overcome these issues [8, 13].

An issue that is perhaps less discussed in ensemble filtering literature is the inclusion of model error in the methods. In many methods, the only straightforward way to include model error is by randomly perturbing the forecast ensemble. The model error covariance matrix often cannot be directly inserted into the filtering equations, since it would prevent the memory and CPU efficient low-rank representation of the matrices which makes ensemble filters computationally attractive. The underestimation of the uncertainty in the state estimates due to missing model error terms is often compensated by introducing covariance inflation.

In this paper, we propose an ensemble filtering method, where new samples are directly drawn from the posterior distribution. For the sampling, we use an optimization-based technique called *randomize then optimize* (RTO) that is based on repeatedly minimizing a randomly perturbed negative log-posterior cost function [14]. If the observation operator is linear, we can show that the method yields exact samples from the Gaussian approximation to the posterior, but the method can also be implemented for nonlinear observation models. In the proposed method, the model error covariance matrix is included directly in the minimized cost function, which yields a full-rank forecast error covariance matrix. Including the model error term has a regularizing effect on the forecast error covariance, and the problem of spurious correlations (and thus the need for localization) is diminished. Moreover, possible covariance inflation can be directly implemented via tuning of the model error covariance matrix.

The developed method is close to the recently introduced variational ensemble Kalman filtering (VEnKF) ideas [15, 16]. In VEnKF, the posterior estimate is computed by solving an optimization problem, and an approximation of the posterior covariance is obtained from the search path of the optimizer. The new ensemble is then generated using the covariance approximations. In the method proposed in this paper, no approximations are made and the new ensemble is drawn exactly from the Gaussian approximation to the posterior, assuming that the observation model is linear. The method is computationally more challenging, since it involves solving a large-scale optimization task many times. However, the optimizations needed in ensemble generation can be easily parallelized.

The RTO approach is, in principle, close to the Ensemble of Data Assimilations (EDA) approach developed at ECMWF [17], where multiple 4D-Var optimization problems are solved with perturbed observations and perturbed background (prior) states to quantify the uncertainty in the state estimates. Here, the approach is developed and tested in an ensemble Kalman filtering context.

The optimization-based ensemble filter is tested with two benchmark models: a 40-dimensional Lorenz model and 1600 dimensional two-layer quasi-geostrophic model. In both cases, the method performs much better than the classical EnKF, especially with small ensemble size when the rank-deficiency problem in EnKF is highlighted.

The paper is organized as follows. In Section 2 we briefly review the basics of ensemble Kalman filtering, and in Section 3 we present our optimization-based approach. Section 4 is reserved for discussion and specific remarks related to the method. Section 5 presents the numerical examples and Section 6 concludes the paper.

2. ENSEMBLE KALMAN FILTERING

Let us start by considering the problem of estimating the state $\mathbf{x}_k$ of a nonlinear dynamical model $\mathcal{M}$ at discrete times k based on observations $\mathbf{y}_k$, when the observation model is linear:

$$\mathbf{x}_k = \mathcal{M}(\mathbf{x}_{k-1}) + \mathbf{q}_k \quad (1)$$
$$\mathbf{y}_k = \mathbf{K}_k\mathbf{x}_k + \mathbf{r}_k. \quad (2)$$

In the above system, $\mathbf{K}_k$ is the $m \times d$ observation operator, the $d \times 1$ vector $\mathbf{x}_k$ represents the model state, and observed data are denoted by the $m \times 1$ vector $\mathbf{y}_k$. Model error $\mathbf{q}_k$ and observation error $\mathbf{r}_k$ are assumed to be normally distributed zero mean random vectors with covariance matrices $\mathbf{Q}_k$ and $\mathbf{R}_k$, respectively.

The standard method of solving the above problem is the extended Kalman filter (EKF), where the nonlinear model is linearized, $\mathbf{M}_k = \partial\mathcal{M}(\mathbf{x}^{\text{est}}_{k-1})/\partial\mathbf{x}$, and the Kalman filter formulas are applied. Using our notation, the EKF algorithm is can be written as follows:

The extended Kalman filter algorithm, given $\mathbf{x}^{\text{est}}_0$, $\mathbf{C}^{\text{est}}_0$, *and* $k = 1$.

1. Move the state estimate and covariance in time:
   a. Compute the prior mean $\mathbf{x}^p_k = \mathcal{M}(\mathbf{x}^{\text{est}}_{k-1})$.
   b. Compute the prior covariance $\mathbf{C}^p_k = \mathbf{M}_k\mathbf{C}^{\text{est}}_{k-1}\mathbf{M}^{\text{T}}_k + \mathbf{Q}_k$.
2. Combine the prior with observations:
   a. Compute the Kalman gain $\mathbf{G}_k = \mathbf{C}^p_k\mathbf{K}^{\text{T}}_k(\mathbf{K}_k\mathbf{C}^p_k\mathbf{K}^{\text{T}}_k + \mathbf{R}_k)^{-1}$.
   b. Compute the state estimate $\mathbf{x}^{\text{est}}_k = \mathbf{x}^p_k + \mathbf{G}_k(\mathbf{y}_k - \mathbf{K}_k\mathbf{x}^p_k)$.
   c. Compute the covariance estimate $\mathbf{C}^{\text{est}}_k = \mathbf{C}^p_k - \mathbf{G}_k\mathbf{K}_k\mathbf{C}^p_k$.
3. Set $k \to k + 1$ and go to step i.

In high-dimensional problems, the computational requirements of EKF become prohibitively high; even storing the covariance matrices needed in the computations can be infeasible. In ensemble filtering, the uncertainty in the state estimate $\mathbf{x}_k$ is represented as N samples, here denoted as $\mathbf{s}_k = (\mathbf{s}_{k,1}, \mathbf{s}_{k,2}, ..., \mathbf{s}_{k,N})$, instead of a covariance matrix. The ensemble Kalman filter (EnKF, [4, 12]) essentially replaces the state covariance matrices in KF with the sample covariance calculated from the ensemble. The sample covariance can be written as $\text{Cov}(\mathbf{s}_k) = \mathbf{X}_k\mathbf{X}^{\text{T}}_k$, where

$$\mathbf{X}_k = ((\mathbf{s}_{k,1} - \overline{\mathbf{s}_k}), (\mathbf{s}_{k,2} - \overline{\mathbf{s}_k}), ..., (\mathbf{s}_{k,N} - \overline{\mathbf{s}_k}))\,/\sqrt{N-1}. \quad (3)$$

The sample mean is denoted by $\overline{\mathbf{s}_k}$. Using our notation, the EnKF algorithm can be formulated as follows.

The ensemble Kalman filter algorithm, given $\mathbf{s}^{\text{est}}_0$ *and* $k = 1$.

1. Move the state estimate and covariance in time:
   a. Move ensemble forward and perturb members with model error:
   $\mathbf{s}^p_{k,i} = \mathcal{M}(\mathbf{s}^{\text{est}}_{(k-1),i}) + \mathbf{q}_{k,i},\ i = 1, ..., N.$
   b. Calculate sample mean $\overline{\mathbf{s}_k}$ and covariance $\mathbf{C}^p_k = \mathbf{X}_k\mathbf{X}^{\text{T}}_k$.
2. Combine the prior with observations:
   a. Compute the Kalman gain $\mathbf{G}_k$.
   b. Update ensemble members $\mathbf{s}^{\text{est}}_{k,i} = \mathbf{s}^p_{k,i} + \mathbf{G}_k(\mathbf{y}_k - \mathbf{K}_k\mathbf{s}^p_{k,i} + \mathbf{r}_{k,i})$.
   c. Calculate state estimate as the sample mean $\overline{\mathbf{s}^{\text{est}}_{k,i}}$.
3. Set $k \to k + 1$ and go to step i.

In the above algorithm, vectors $\mathbf{q}_{k,i}$ and $\mathbf{r}_{k,i}$ are realizations of the model error and observation error distributions, which are Gaussians with covariances $\mathbf{Q}_k$ and $\mathbf{R}_k$, respectively.

The EnKF is easy to implement and also computationally feasible for large-scale models. This is due to the fact that the prior covariance $\mathbf{C}_k^p$ in the above algorithm can be kept in a "low-rank" ensemble form $\mathbf{X}_k\mathbf{X}_k^{\mathrm{T}}$. However, at the same time, one could criticize that the analysis is restricted to the subspace of the state space spanned by the prior ensemble. The rank-deficiency of the prior covariance matrix can cause many problems, such as ensemble inbreeding, introduction of spurious correlations, and underestimation of the covariance matrix. To overcome these issues, different localization (covariance tapering) and covariance inflation techniques can be applied. In covariance tapering, the prior covariance matrix is multiplied element-wise with a tapering matrix that cuts off the unphysical long distance correlations; see [18, 19]. In covariance inflation, the prior covariance matrix is artificially enlarged to avoid filter divergence due to too narrow prior covariance matrix.

Recently, several other ensemble filtering methods have been developed to overcome some issues in the standard EnKF. For instance, the so-called square root ensemble filters, see [5–7], make a deterministic transformation from the prior ensemble into the posterior ensemble so that the posterior sample statistics match with theory. In square root filtering, no perturbed observations are used, and the methods can have more favorable error scaling as a function of ensemble size. However, the square root filters also operate with rank-deficient matrices and the possible problems of ensemble inbreeding, covariance underestimation, spurious correlations, and including model error remain. In our optimization-based ensemble filter introduced next, the model error is included directly as a covariance matrix, which regularizes the sample covariance matrix and diminishes spurious correlations; see Section 4 for a small-scale demonstration. Moreover, covariance inflation can be obtained via tuning the model error covariance matrix. In the proposed method, the new ensemble is always randomly resampled from the posterior in the full state space, which can avoid the problem of ensemble inbreeding.

3. OPTIMIZATION-BASED SAMPLING IN ENKF

Combining the previous forecast with the new observations yields the following probability density for the state at time k:

$$\pi(\mathbf{x}_k) \propto \exp\left(-\frac{1}{2}||\mathbf{y}_k - \mathbf{K}_k\mathbf{x}_k||^2_{\mathbf{R}_k} - \frac{1}{2}||\mathbf{x}_k - \mathbf{x}_k^p||^2_{\mathbf{C}_k^p}\right), \tag{4}$$

where the notation $||\mathbf{x}||^2_{\mathbf{B}} = \mathbf{x}^T\mathbf{B}^{-1}\mathbf{x}$. In the case of a linear observation operator as above, we can use the *randomize then optimize* (RTO) procedure, which has been successfully used in statistical analysis of inverse problems [14], to get independent samples from $\pi(\mathbf{x}_k)$.

The RTO technique for sampling from a density $p(\mathbf{x}) \propto \exp(-0.5||\mathbf{b} - \mathbf{A}\mathbf{x}||^2_{\mathbf{C}})$ works as follows: generate random data $\hat{\mathbf{b}} \sim \mathrm{N}(\mathbf{b}, \mathbf{C})$ and then repeatedly solve the linear least squares problem $\hat{\mathbf{x}} = \arg\min ||\hat{\mathbf{b}} - \mathbf{A}\mathbf{x}||^2_{\mathbf{C}}$. It is straightforward to verify that the RTO approach produces samples from the correct density. If the model is linear, the least squares solution can be written down analytically: $\hat{\mathbf{x}} = (\mathbf{A}^T\mathbf{C}^{-1}\mathbf{A})^{-1}\mathbf{A}^T\mathbf{C}^{-1}\hat{\mathbf{b}} = \mathbf{P}\hat{\mathbf{b}}$; see, e.g., [20]. That is, we have a linear mapping from the distribution of the data to the distribution of the LSQ solution, and we can write down the mean and covariance matrix of the solution using the basic formulas $\mathrm{E}(\mathbf{P}\hat{\mathbf{b}}) = \mathbf{P}\mathrm{E}(\hat{\mathbf{b}})$ and $\mathrm{Cov}(\mathbf{P}\hat{\mathbf{b}}) = \mathbf{P}\mathrm{Cov}(\hat{\mathbf{b}})\mathbf{P}^T$:

$$\mathrm{E}(\hat{\mathbf{x}}) = (\mathbf{A}^T\mathbf{C}^{-1}\mathbf{A})^{-1}\mathbf{A}^T\mathbf{C}^{-1}\mathbf{b} \tag{5}$$

$$\mathrm{Cov}(\hat{\mathbf{x}}) = (\mathbf{A}^T\mathbf{C}^{-1}\mathbf{A})^{-1}. \tag{6}$$

These are the well-known formulas for the mean and the covariance of the Gaussian density $p(\mathbf{x})$ defined above; that is, by repeatedly generating new data and solving the resulting LSQ problem, we obtain samples from the correct density.

Returning to the original density of interest (4), we can write $\pi(\mathbf{x})$ in the same form as $p(\mathbf{x})$ by defining

$$\mathbf{A} = \begin{bmatrix} \mathbf{K}_k \\ \mathbf{I} \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} \mathbf{y}_k \\ \mathbf{x}_k^p \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} \mathbf{R}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_k^p \end{bmatrix}. \tag{7}$$

To use RTO for generating a sample from $\pi(\mathbf{x})$, we generate new data $\hat{\mathbf{y}}_k \sim \mathrm{N}(\mathbf{y}_k, \mathbf{R}_k)$ and a new prior mean $\hat{\mathbf{x}}_k^p \sim \mathrm{N}(\mathbf{x}_k^p, \mathbf{C}_k^p)$, and then minimize the quadratic expression

$$||\hat{\mathbf{b}} - \mathbf{A}\mathbf{x}||_{\mathbf{C}}^2 = ||\hat{\mathbf{y}}_k - \mathbf{K}_k\mathbf{x}_k||_{\mathbf{R}_k}^2 + ||\mathbf{x}_k - \hat{\mathbf{x}}_k^p||_{\mathbf{C}_k^p}^2. \tag{8}$$

The form of the prior covariance matrix $\mathbf{C}_k^p$ depends on the data assimilation method applied. In 3D-Var we would have $\mathbf{C}_k^p = \mathbf{B}_k$, where $\mathbf{B}_k$ is the background covariance matrix given directly by the user without propagating the covariance at the previous time step forward. For EKF, $\mathbf{C}_k^p = \mathbf{M}_k\mathbf{C}_{k-1}^{est}\mathbf{M}_k^T + \mathbf{Q}_k$. For our ensemble Kalman filter implementation, we replace the "dynamical" part of the covariance matrix with the empirical covariance estimate, and define

$$\mathbf{C}_k^p = \mathbf{X}_k\mathbf{X}_k^T + \mathbf{Q}_k, \tag{9}$$

where

$$\mathbf{X}_k = ((\mathbf{s}_{k,1} - \mathbf{x}_k^p), (\mathbf{s}_{k,2} - \mathbf{x}_k^p), ..., (\mathbf{s}_{k,N} - \mathbf{x}_k^p))/\sqrt{N}. \tag{10}$$

Here, the ensemble members are not randomly perturbed with the model error as in EnKF, since the model error is included directly as a matrix as in EKF. That is, the ensemble members are $\mathbf{s}_{k,i}^p = \mathcal{M}(\mathbf{s}_{(k-1),i}^{\mathrm{est}})$. Note also that the prior mean is taken to be the prediction $\mathbf{x}_k^p$ as in EKF and not the ensemble mean as in EnKF.

In order to implement RTO sampling, we need to be able to produce samples from normal distributions with covariance matrices $\mathbf{R}_k$ and $\mathbf{C}_k^p$. That is, we need to compute symmetric decompositions (square roots) of the matrices. In practice, the measurement error covariance matrix $\mathbf{R}_k$ is often a (block) diagonal matrix, and it should be feasible to compute, e.g., the Cholesky decomposition of the matrix. For the prior covariance matrix, the "ensemble part" $\mathbf{X}_k\mathbf{X}_k^T$ is directly in a square form, and only the square root of $\mathbf{Q}_k$, denoted by $\mathbf{Q}_k^{1/2}$, is needed. Then, a random sample $\hat{\mathbf{x}}$ from a zero centered normal distribution with covariance matrix $\mathbf{C}_k^p$ can be computed as follows:

$$\hat{\mathbf{x}} = \mathbf{Q}_k^{1/2}\mathbf{z}_d + \mathbf{X}_k\mathbf{z}_N, \tag{11}$$

where d is the dimension of the state space, N is the number of samples used in the ensemble Kalman filter, and $\mathbf{z}_d$ denotes a d-dimensional standard normal vector.

Moreover, if iterative optimization methods are used in the generation of the new ensemble, we need to be able to evaluate the quadratic expression (8). The likelihood term is usually straightforward, since the measurement error covariance matrices $\mathbf{R}_k$ are often (block) diagonal. For the prior term, we can apply the matrix inversion lemma, see [21], which yields

$$\begin{aligned} (\mathbf{C}_k^p)^{-1} &= (\mathbf{X}_k\mathbf{X}_k^{\mathrm{T}} + \mathbf{Q}_k)^{-1} && (12)\\ &= \mathbf{Q}_k^{-1} - \mathbf{Q}_k^{-1}\mathbf{X}_k(\mathbf{I} + \mathbf{X}_k^{\mathrm{T}}\mathbf{Q}_k^{-1}\mathbf{X}_k)^{-1}\mathbf{X}_k^{\mathrm{T}}\mathbf{Q}_k^{-1}. && (13)\end{aligned}$$

If the form of $\mathbf{Q}_k$ is simple (e.g., block diagonal), $\mathbf{Q}_k^{-1}$ can be computed and the above expression can be used when evaluating (8) without handling full dense matrices of size $d \times d$. Note that the inversion left in the above formula is only for a $N \times N$ matrix, where N is the ensemble size, which is typically limited to a small number.

Finally, the optimization-based ensemble Kalman filter can be formulated as an algorithm as follows:

The RTO ensemble Kalman filter algorithm, given $\mathbf{s}_0^{\mathrm{est}}$ *and* $k = 1$.

1. Move the state estimate and ensemble in time:
   a. Move the state estimate forward: $\mathbf{x}_k^p = \mathcal{M}(\mathbf{x}_{k-1}^{\mathrm{est}})$.
   b. Move ensemble forward: $\mathbf{s}_{k,i}^p = \mathcal{M}(\mathbf{s}_{(k-1),i}^{\mathrm{est}}),\ i = 1, ..., N$.
   c. Define $\mathbf{C}_k^p = \mathbf{X}_k\mathbf{X}_k^T + \mathbf{Q}_k$, where $\mathbf{X}_k$ is given by (10).
2. Combine the prior with observations:

a. Compute the state estimate $\mathbf{x}_k^{\mathrm{est}}$ by minimizing $||\mathbf{y}_k - \mathbf{K}_k\mathbf{x}_k||^2_{\mathbf{R}_k} + ||\mathbf{x}_k - \mathbf{x}_k^p||^2_{\mathbf{C}_k^p}$.

b. Sample the new ensemble $\mathbf{s}_{k,i}^{\mathrm{est}}$ by repeatedly generating $\hat{\mathbf{y}}_k \sim \mathrm{N}(\mathbf{y}_k, \mathbf{R}_k)$ and $\hat{\mathbf{x}}_k^p \sim \mathrm{N}(\mathbf{x}_k^p, \mathbf{C}_k^p)$ and minimizing $||\hat{\mathbf{y}}_k - \mathbf{K}_k\mathbf{x}_k||^2_{\mathbf{R}_k} + ||\mathbf{x}_k - \hat{\mathbf{x}}_k^p||^2_{\mathbf{C}_k^p}$.

3. Set $k \to k+1$ and go to step i.

Next, we give some specific remarks and discussion related to the presented algorithm.

4. REMARKS

Relation to EnKF. The proposed method is close to the original ensemble Kalman filter of [4], where the Kalman filter formulas are repeatedly applied to perturbed observations and perturbed model forecasts. A difference between the approaches is that in EnKF the prior covariance matrix is given in the subspace spanned by the ensemble, $\mathbf{C}_k^p = \tilde{\mathbf{X}}_k\tilde{\mathbf{X}}_k^T$, where $\tilde{\mathbf{X}}_k$ are the propagated samples perturbed with random draws from the model error distribution. In our method, the prior covariance matrix is $\mathbf{C}_k^p = \mathbf{Q}_k + \mathbf{X}_k\mathbf{X}_k^T$, where $\mathbf{X}_k$ are the unperturbed ensemble members. As we see in our examples, this small difference can have a significant effect on the performance of the method. Another difference to EnKF is that the prior mean is taken to be the predicted state estimate as in EKF, not the mean of the predicted ensemble. According to our experience, this difference has a smaller impact on the performance.

Note that we could also plug in the alternative forecast error covariance matrix formulation $\mathbf{C}_k^p = \mathbf{Q}_k + \mathbf{X}_k\mathbf{X}_k^T$ directly to the Kalman filter formulas, and generate the new ensemble from the resulting Gaussian posterior. This would give equivalent results compared to RTO-EnKF; the optimization step would be carried out with direct formulas instead of numerical optimization routines. However, this is applicable only in small-scale problems, since one would need to operate with full $d \times d$ covariance matrices. Here, the RTO technique provides an efficient way to produce samples from the high-dimensional Gaussian distributions without having to handle full covariance matrices.

Spurious Correlations and Localization. As mentioned in Section 2, many ensemble methods suffer from physically unrealistic correlations which are caused by rank-deficient covariance matrices computed using a small number of samples. Different covariance tapering (or localization) techniques are often used to cut off these correlations, for instance, by taking the element-wise product of the sample covariance matrices and compactly supported covariance functions [19]. We note that in RTO-EnKF, the problem of spurious correlations is diminished. This is due to the above mentioned difference in the way the prior covariance matrix is defined: in the proposed method $\mathbf{C}_k^p = \mathbf{Q}_k + \mathbf{X}_k\mathbf{X}_k^T$, whereas in standard EnKF (and many other ensemble filters) $\mathbf{C}_k^p = \tilde{\mathbf{X}}_k\tilde{\mathbf{X}}_k^T$. That is, the model error covariance has a regularizing effect on $\mathbf{C}_k^p$. The two ways of computing the prior covariance matrix are illustrated in Fig. 1, where the correlation matrices at one time instant are compared in the Lorenz toy model case described in Section 5.1. One can clearly see the difference in the amount of strong correlations between spatially distant points; off-diagonal entries are overall significantly higher for EnKF.

Although the need for localization in RTO-EnKF is diminished, a well implemented localization technique is likely beneficial for RTO-EnKF as well. Inflation can be included in RTO-EnKF by using a tapering matrix that introduces sparsity to the prior covariance matrix. More specifically, the tapered prior covariance for RTO-EnKF reads as $\mathbf{C}_k^p = \rho \circ (\mathbf{X}_k\mathbf{X}_k^T) + \mathbf{Q}_k$, where ρ is the tapering matrix and $\circ$ denotes the element-wise product. When tapering is used, the rank of the matrix $\rho \circ (\mathbf{X}_k\mathbf{X}_k^T)$ is increased, and the matrix inversion lemma (13) is no longer applicable for computing the prior term in the cost function efficiently. However, if a sparse ρ is applied and if $\mathbf{Q}_k$ is simple (e.g., block diagonal), the matrix $\mathbf{C}_k^p$ becomes sparse. This allows us to perform, for instance, a Cholesky factorization that takes advantage of the sparsity: $\mathbf{C}_k^p = \mathbf{L}_k\mathbf{L}_k^T$, where $\mathbf{L}_k$ is sparse. Now, the prior term in the optimized cost function can be computed as

$$\mathbf{r}_k^T(\rho \circ (\mathbf{X}_k\mathbf{X}_k^T) + \mathbf{Q}_k)^{-1}\mathbf{r}_k = \mathbf{r}_k^T(\mathbf{L}_k\mathbf{L}_k^T)^{-1}\mathbf{r}_k = (\mathbf{L}_k^{-1}\mathbf{r}_k)^T(\mathbf{L}_k^{-1}\mathbf{r}_k), \tag{14}$$

where $\mathbf{r}_k = \mathbf{x}_k - \mathbf{x}_k^p$. Sparsity introduced by tapering was also used to speed up matrix computations in ensemble Kalman filtering and smoothing in [22].

Localization often improves the performance of ensemble Kalman filters remarkably; see Section 5 for a numerical example. However, building and tuning the tapering matrix can be complicated; for instance, the distance between

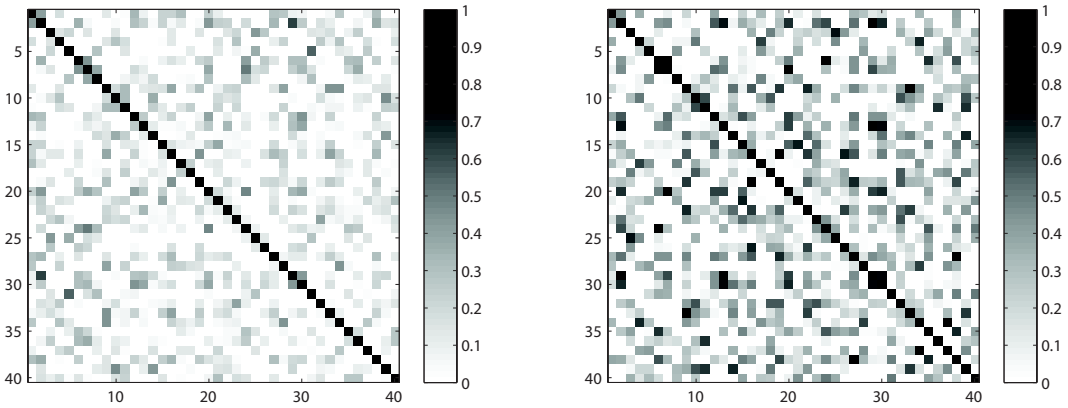

FIG. 1: Correlation matrices corresponding to $\mathbf{C}_k^p$ at one time instant of the Lorenz toy model in the RTO-EnKF (left) and classical EnKF (right) with ensemble size $N = 10$.

different types of state variables can be unclear. Moreover, localization assumes that one has a natural ordering of the variables available in terms of distance. If this is lacking, localization can be difficult to implement. We note again that RTO-EnKF already contains regularization for the forecast covariance matrix in the form of $\mathbf{Q}_k$, and thus the localization is less crucial.

Tuning the model error covariance matrix. The model error covariance matrix $\mathbf{Q}_k$ is often considered as a calibration parameter for filtering methods. Here, we do not consider the tuning of $\mathbf{Q}_k$ in detail, but point the reader to [23] and [24] that discuss the estimation of parameterized model error covariance matrices in filtering methods. We also note that the $\mathbf{Q}_k$ that yields optimal filter accuracy should not contain only "model error," but also imperfections in other parts of the assimilation system (for instance, errors related to small ensemble size). Fortunately, in many cases, including our numerical examples in this paper, a rather broad range of $\mathbf{Q}_k$ matrices result in acceptable filter performance in terms of accuracy of the state estimates.

Nonlinear observation operators. The RTO sampling procedure can be carried out with nonlinear observation operators $\mathcal{K}(\mathbf{x})$, which makes the target density (4) non-Gaussian, and the resulting optimization problem non-quadratic. In the nonlinear case, however, it is less clear what the density is that RTO samples from. In our recent research [26] we have found the density for the nonlinear case, which makes it possible to correct the RTO samples towards the non-Gaussian posterior using Metropolis-Hastings and importance sampling schemes. However, our experience with RTO sampling in non-Gaussian static estimation problems suggests that the RTO yields a rather good approximation to the posterior even without such corrections.

CPU and implementation issues. In small and moderate dimensional linear problems, we can solve the minimization exactly by solving a linear system of equations. In large-scale cases and nonlinear cases, we need to resort to iterative optimization methods, such as conjugate gradient or quasi-Newton algorithms. The computational cost of the method is larger compared to many other ensemble methods, since it involves the solution of many large-scale optimization problems. However, the cost function does not include evaluations of the forward model, which is usually the computational bottleneck in large-scale data assimilation. The dominant cost in the optimization is matrix-vector multiplication. Moreover, if the observation model is linear, the optimization problems are quadratic, for which efficient iterative algorithms exist, such as limited memory BFGS and conjugate gradient (CG) methods. These methods are guaranteed to converge for quadratic problems with positive definite Hessian. Their convergence rate is determined by the eigenvalues of the Hessian and hence improved convergence rates can be obtained via preconditioning. These algorithms can converge very fast; for instance, for the 1600-dimensional benchmark problem given in Section 5, around 10 iterations of the limited-memory BFGS algorithm was sufficient to obtain a posterior sample. In addition, the optimization tasks for ensemble generation are trivially parallelizable.

Including additional priors. A benefit of the variational formulation of the proposed approach compared to many other ensemble methods is that it is rather straightforward to include prior information about the state vector,

for instance, related to the smoothness of the spatial fields. In particular, adding Gaussian priors is easy: with prior mean $\mathbf{m}$ and covariance matrix $\mathbf{L}$, the cost function reads as

$$||\mathbf{y}_k - \mathbf{K}_k\mathbf{x}_k||^2_{\mathbf{R}_k} + ||\mathbf{x}_k - \mathbf{x}_k^p||^2_{\mathbf{C}_k^p} + ||\mathbf{x}_k - \mathbf{m}||^2_{\mathbf{L}}, \tag{15}$$

and in RTO sampling one simply has to perturb $\mathbf{m}$ with random numbers drawn from $\mathrm{N}(\mathbf{0}, \mathbf{L})$ every time the cost function is optimized to get a new sample. Including such priors can be problematic in many other ensemble filtering techniques.

Applicability in variational data assimilation, connection to EDA. As mentioned before, the approach can be directly used within 3D-Var to obtain samples from the analysis densities, which can in turn be useful in uncertainty quantification. Optimization-based RTO sampling can also be used within 4D-Var methods. For the strong-constraint 4D-Var, the target density at time step k can be written as follows:

$$\pi_{\mathrm{4d}}(\mathbf{x}_k) \propto \exp\left(-\frac{1}{2}||\mathbf{x}_k - \mathbf{x}_k^p||^2_{\mathbf{B}_k} - \frac{1}{2}\sum_{i=1}^{T}||\mathbf{y}_{k+i} - \mathcal{K}(\mathcal{M}_i(\mathbf{x}_k))||^2_{\mathbf{R}_{k+i}}\right), \tag{16}$$

where $\mathcal{M}_i(\mathbf{x}_k)$ denotes the evolution model propagated forward i steps from the initial state $\mathbf{x}_k$. The use of RTO sampling in this case would correspond to generating the prior means $\hat{\mathbf{x}}_k^p \sim \mathrm{N}(\mathbf{x}_k^p, \mathbf{B}_k)$ and data $\hat{\mathbf{y}}_{k+i} \sim \mathrm{N}(\mathbf{y}_{k+i}, \mathbf{R}_{k+i})$ and maximizing the above expression using the generated $\hat{\mathbf{x}}_k^p$ and $\hat{\mathbf{y}}_{k+i}$. Note that this optimization task is no longer quadratic and would yield samples from a non-Gaussian density.

Similarly, the approach could be extended to weak-constraint 4D-Var, where the model is not assumed to be perfect within the assimilation window and is allowed to make jumps:

$$\begin{aligned}\pi_{\mathrm{w4d}}(\mathbf{x}_k, \mathbf{x}_{k+1}, ..., \mathbf{x}_{k+T}) \propto \exp\Bigg(&-\frac{1}{2}||\mathbf{x}_k - \mathbf{x}_k^p||^2_{\mathbf{B}_k} - \frac{1}{2}\sum_{i=1}^{T}||\mathbf{y}_{k+i} - \mathcal{K}(\mathcal{M}_i(\mathbf{x}_k))||^2_{\mathbf{R}_{k+i}} \\ &- \frac{1}{2}\sum_{i=1}^{T}||\mathbf{q}_{k+i} + \mathbf{q}_\mu||^2_{\mathbf{Q}_{k+i}}\Bigg),\end{aligned}$$

where $\mathbf{q}_{k+i} = \mathbf{x}_{k+i} - \mathcal{M}_1(\mathbf{x}_{k+i-1})$ are the model jumps, $\mathbf{q}_\mu$ is the model error mean and $\mathbf{Q}_{k+i}$ are the model error covariance matrices. Here, in addition to perturbing $\mathbf{x}_k^p$ and $\mathbf{y}_{k+i}$, one would generate the model error means $\hat{\mathbf{q}}_\mu \sim \mathrm{N}(\mathbf{q}_\mu, \mathbf{Q}_{k+i})$. This would yield samples from the joint distribution of $(\mathbf{x}_k, \mathbf{x}_{k+1}, ..., \mathbf{x}_{k+T})$, from which one can obtain samples from the marginal densities: the last index $k+T$ corresponds to the filtering distribution and the intermediate indices are essentially smoothing distributions, as discussed in [26].

A similar approach called Ensemble of Data Assimilations (EDA) is developed and applied to operational NWP at the European Center of Medium Range Weather Forecasts (ECMWF); see [17]. In EDA, perturbed observations and forecasts are used to generate an ensemble of 4D-Var data analyses, with the purpose of estimating the analysis uncertainty; generate more realistic initial state perturbations for ensemble prediction systems, and introduce flow dependency in 4D-Var background covariance matrices [27]. The difference to our approach is that we formulate the approach in the EnKF context, whereas EDA is is based on 4D-Var.

Possible pitfalls. A few pitfalls remain in the proposed approach. First of all, the method assumes that the repeated optimization tasks can be solved accurately; if the optimizers do not converge, the quality of the obtained samples can be poor. Moreover, many other ensemble filtering methods allow assimilating observations in batches; see [7, 11], which is handy in high-dimensional problems, especially if the number of assimilated observations is high. In the proposed method, all observations are assimilated in one step, and it is not clear if processing observations in batches is possible.

5. NUMERICAL EXAMPLES

In this section, we compare the proposed method to the standard EnKF in two synthetic examples. The first example is the well-known Lorenz '96 benchmark problem (a 40-dimensional nonlinear chaotic ODE system), that shares some

characteristics with weather models [28, 29]. The second example is a 1600-dimensional two-layer quasi-geostrophic model, which is often used in data assimilation benchmarking; see, e.g., [30].

For comparing methods, we use the root mean squared (RMS) error in the state estimate, written as

$$\mathrm{RMS}_k = \sqrt{\frac{1}{d}||\mathbf{x}_k^{\mathrm{est}} - \mathbf{x}_k^{\mathrm{true}}||^2} \tag{17}$$

where, at iteration k, $\mathbf{x}_k^{\mathrm{est}}$ is the filter estimate, $\mathbf{x}_k^{\mathrm{true}}$ is the truth used in data generation, and d is the dimension of the state vector. While the RMS error indicates how well a method is able to estimate the mean of the state, it does not tell anything about the quality of the uncertainty estimates (e.g., variances) derived from the ensemble. It is known that EnKF methods can perform quite poorly in capturing this uncertainty [31], and we do not claim that RTO-EnKF is any better in this respect. The quality of the ensemble methods in quantifying the uncertainty can be monitored via techniques like rank histograms and ranked probability scores [32, 33]. Here, we do not study how the filters behave in this sense, and focus only on the accuracy of the mean.

Here, the goal is to demonstrate that the rank deficiency problems are diminished in the RTO-EnKF approach. That is, we do not comprehensively discuss, for instance, different covariance localization techniques that can significantly improve the performance of ensemble filters. However, we do compare the methods with localization included and show that RTO-EnKF can also benefit from localization.

5.1 Lorenz '96

In this example, we consider the well-known nonlinear and chaotic Lorenz '96 model [28, 29]. The model shares many characteristics with realistic atmospheric models and it is often used as a low-order test case for data assimilation schemes. We use a 40-dimensional version of the model, given as an ODE system as

$$\frac{dx_i}{dt} = (x_{i+1} - x_{i-2})x_{i-1} - x_i + 8, \quad i = 1, 2, ..., 40. \tag{18}$$

The state variables are periodic: $x_{-1} = x_{39}$, $x_0 = x_{40}$ and $x_{41} = x_1$. Out of the 40 model states, measurements are obtained from 24 states. We define the observation operator as $\mathcal{K}(\mathbf{x}) = \mathbf{K}\mathbf{x}$, where

$$[\mathbf{K}]_{rp} = \begin{cases} 1, & (r,p) \in \{(3j+i, 5j+i+2)\} \\ 0 & \text{otherwise} \end{cases} \tag{19}$$

where $i = 1, 2, 3$ and $j = 0, 1, ..., 7$. Thus, we observe the last three states in every set of five. To generate data, we add Gaussian noise to the model solution with zero mean and covariance matrix $(0.15\sigma_{\mathrm{clim}})^2\mathbf{I}$, where $\sigma_{\mathrm{clim}} = 3.641$ (climatological standard deviation). In the filtering methods, we use $\mathbf{Q}_k = (0.1\sigma_{\mathrm{clim}})^2\mathbf{I}$ as the model error covariance matrix and $\mathbf{R}_k = (0.15\sigma_{\mathrm{clim}})^2\mathbf{I}$ as the observation error covariance matrix. As initial guesses in the filtering, we use $\mathbf{x}_0^{\mathrm{est}} = \mathbf{1}$ and $\mathbf{C}_0^{\mathrm{est}} = \mathbf{I}$.

The comparison of the relative error for the EnKF and RTO-EnKF filter estimates is given in Fig. 2 for ensemble sizes $N = 10$ and $N = 20$. One can see that the performance of the RTO-EnKF is significantly better that that of the standard EnKF, and with $N = 10$ EnKF fails to converge. With ensemble sizes $N \geq 30$, both filters work, roughly speaking, in a comparable way. That is, the benefit of RTO-EnKF is seen especially with small ensemble sizes, which was expected, since the problems caused by rank deficiency get worse as the ensemble size gets smaller.

The above comparison is not fair in the sense that a localization technique would likely improve the performance of EnKF. However, it does suggest that in absence of localization, RTO-EnKF has some desirable properties. We also compared the methods with localization included. We constructed the tapering matrix using the popular the fifth-order piecewise rational function [19], where the correlation cutoff length was chosen experimentally so that good filter performance was obtained. In Fig. 3, we compare the RMS errors of the state estimates with $N = 6$ and $N = 10$. One can see that localization dramatically improves EnKF performance, but also helps RTO-EnKF; with small ensemble sizes, RTO-EnKF still outperforms the standard EnKF.

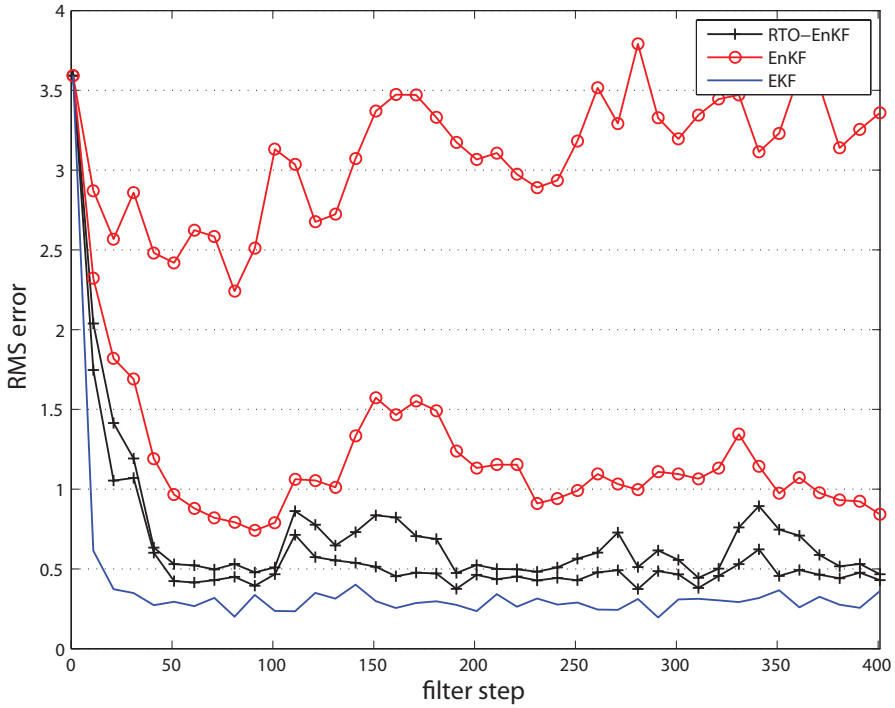

FIG. 2: Comparison of relative error of filter estimates for EKF, EnKF, and RTO-EnKF in the Lorenz '96 model. For the ensemble methods, ensemble sizes $N = 10$ and $N = 20$ were used.

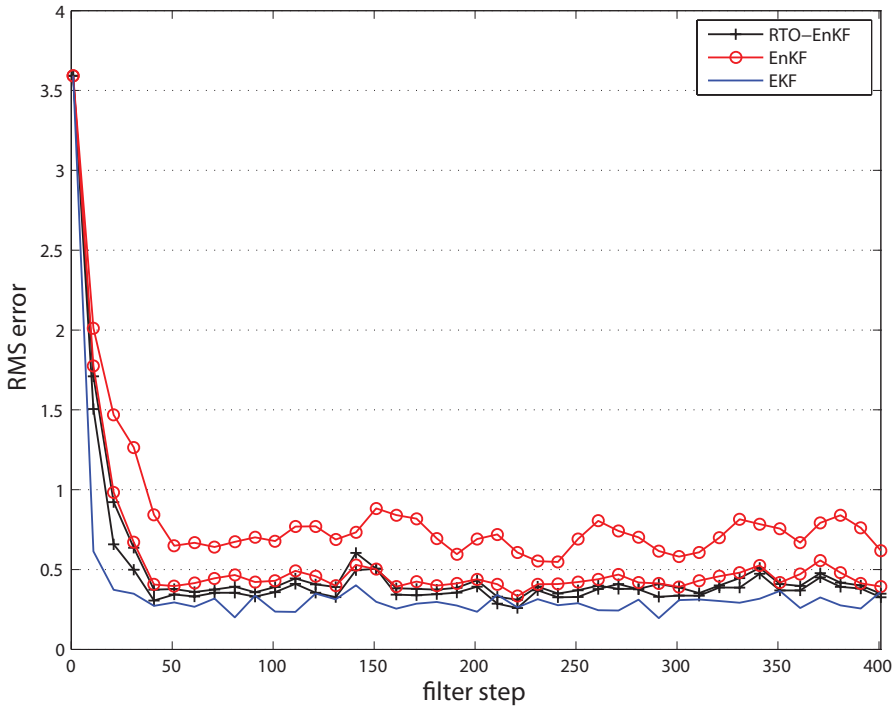

FIG. 3: Comparison of relative error of filter estimates for EKF, EnKF, and RTO-EnKF in the Lorenz '96 model with localization included. For the ensemble methods, ensemble sizes $N = 6$ and $N = 10$ were used.

5.2 Two-Layer Quasi-Geostrophic Model

This part is devoted to the two-layer quasi-geostrophic model (QG model) [34], which serves an example of a large-scale chaotic dynamics in our numerical experiments. We begin by considering the geometry of the QG model and discuss the system of PDE's that define the model dynamics. We also highlight the principles of the numerical solution used to integrate the system. Thereafter, we discuss the settings assigned to the model parameters in our benchmarks and review the obtained results.

5.2.1 Model Description

The two-layer quasi-geostrophic model simulates atmospheric flow for the geostrophic (slow) wind motions. This model can be used as a benchmark for data assimilation in NWP systems as it supports some features common for operational weather models, such as baroclinic instability. At the same time, the QG model has relatively low computational complexity and requires no special hardware to run. The geometrical domain of the model is specified by a cylindrical surface vertically divided into two "atmospheric" layers which can interact through the interface between them. The model also accounts for an orographic component that defines the surface irregularities affecting the bottom layer of the model. The geometric structure of the model dictates periodic latitudinal boundary conditions, whereas the values on the top and the bottom of the cylindrical domain are user-supplied constant values. Aside from the layered structure and boundary conditions, the model parameters comprise the depths of the atmospheric layers, potential temperature change across the layer interface, and the mean potential temperature. When geometrical layout of the two-layer QG model is mapped onto a plane it appears as shown in Fig. 4. In the figure, parameters U_1 and U_2 denote mean zonal flows in the top and the bottom atmospheric layers, respectively. The model formulation we use is dimensionless, where the nondimensionalization is defined by the length scale L, velocity scale U, and the layer depths D_1 and D_2.

The model operates on the terms of potential vorticity and stream function, where the latter one is analogous to pressure. The assumption of quasi-geostrophic motion leads to a coupled system of PDE's (20) describing a conservation law for potential vorticity. Essentially, these equations mean that the amount of inflow coming into an infinitely small volume within an infinitely small time period is compensated by the outflow pouring out of the volume over this time period, so that the flows are balanced. The conservation law is given as

$$\frac{D_1 q_1}{Dt} = 0, \ \frac{D_2 q_2}{Dt} = 0, \tag{20}$$

where D_i denotes the substantial derivatives for latitudinal wind u_i and longitudinal wind v_i, defined as $D_i \cdot / Dt = \partial \cdot / \partial t + u_i(\partial \cdot / \partial x) + v_i(\partial \cdot / \partial y)$; q_i denote the potential vorticity functions; index i specifies the top atmospheric layer ($i = 1$) and the bottom layer ($i = 2$). Interaction between the layers, as well as relation between the potential vorticity q_i and the stream function ψ_i, is modeled by the following system of PDE's:

$$q_1 = \nabla^2 \psi_1 - F_1 (\psi_1 - \psi_2) + \beta y, \tag{21}$$

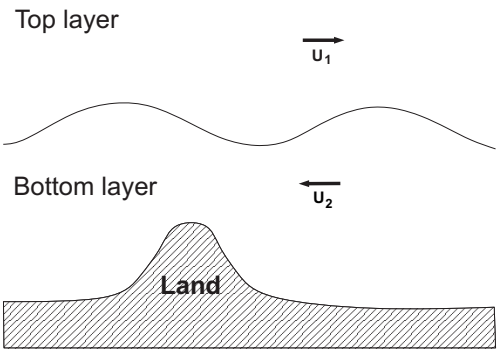

FIG. 4: Geometrical layout of the two-layer Quasi-Geostrophic model.

$$q_2 = \nabla^2\psi_2 - F_2\left(\psi_2 - \psi_1\right) + \beta y + R_s. \tag{22}$$

Here R_s and β denote dimensionless orography component and the northward gradient of the Coriolis parameter, which we hereafter denote as f_0. The relations between the model physic attributes and dimensionless parameters that appear in Eqs. (21)–(22) are as follows:

$$F_1 = \frac{f_0^2 L^2}{\acute{g} D_1},\ F_2 = \frac{f_0^2 L^2}{\acute{g} D_2},\ \acute{g} = g\frac{\Delta\theta}{\bar{\theta}},$$
$$R_s = \frac{S\left(x,y\right)}{\eta D_2},\ \beta = \beta_0 \frac{L}{U},$$

where $\Delta\theta$ defines the potential temperature change across the layer interface, $\bar{\theta}$ is the mean potential temperature, g is acceleration of gravity, $\eta = U/f_0 L$ is the Rossby number associated with the defined system, and $S(x,y)$ and β_0 are dimensional representations of $R_s(x,y)$ and β, respectively.

The system of Eqs. (20)–(22) defines the two-layer quasi-geostrophic model. The the unknown state of the model, and thus the target of estimation, is the stream function ψ_i. For the numerical solution of the system, we consider potential vorticity functions q_1 and q_2 to be known, and invert the spatial Eqs. (21) and (22) for ψ_i. More precisely, we apply ∇^2 to Eq. (21) and subtract F_1 times (22) and F_2 times (21) from the result, which yields the following equation:

$$\begin{aligned}&\nabla^2\left[\nabla^2\psi_1\right] - \left(F_1 + F_2\right)\left[\nabla^2\psi_1\right] \\ &= \nabla^2 q_1 - F_2\left(q_1 - \beta y\right) - F_1\left(q_2 - \beta y - R_s\right).\end{aligned} \tag{23}$$

Equation (23) can be treated as a nonhomogeneous Helmholtz equation with negative parameter $-\left(F_1 + F_2\right)$ and unknown $\nabla^2\psi_1$. Once $\nabla^2\psi_1$ is solved, the stream function for the top atmospheric layer is determined by a Poisson equation. The stream function for the bottom layer can be found by plugging the obtained value for ψ_1 into (21), (22) and solving the equations for ψ_2. The potential vorticity functions q_i are evolved over the time by a numerical advection procedure which models the conservation equations (20). The advection procedure uses linear time interpolation for the speed fields to compute the departure grid points and bicubic spatial interpolation to define potential vorticity values for those points. For more details of the implementation, refer to [35].

5.2.2 Experiment and Results

We run the QG model with 20×40 grid in each layer, and the dimension of the state vector is thus 1600. For generating data, we run the forward model for 50 days with 1 hour time discretization using layer depths $D_1 = 6000$ and $D_2 = 4000$. Data is generated at every 6th step (assimilation step is thus 6 hours) by adding random noise for 100 randomly chosen grid points with standard deviation $\sigma = 2.5 \times 10^{-3}$. For the data assimilation, bias is introduced to the forward model by using wrong layer depths, $\tilde{D}_1 = 5500$ and $\tilde{D}_2 = 4500$. The model error covariance used was $\mathbf{Q} = 0.2\mathbf{I}$. In the RTO optimizations, we applied the limited-memory BFGS optimizer [36]. The quadratic optimizations converged fast in this case; on average, only around 10 L-BFGS iterations were required per sample, which indicates that the approach can be feasible for large-scale problems.

In Fig. 5 we compare the RMS errors of EnKF and RTO-EnKF with different ensemble sizes. For the RTO-EnKF, the smallest ensemble size with which the filter converges and produces reasonable results is roughly $N = 50$. For the standard EnKF without localization, the filter fails to converge with ensemble sizes $N \leq 200$. With a larger ensemble size, $N = 400$, both methods perform in a comparable way. For small ensemble sizes, RTO-EnKF is clearly better: for instance, RTO-EnKF with $N = 50$ performs roughly equally well as EnKF with $N = 300$, and the RTO-EnKF performance with $N = 100$ is similar to EnKF performance with $N = 400$. The results are similar to what was observed with the small-scale Lorenz '96 model, but the difference is even more dramatic. In this larger scale case, the problem of spurious correlations in EnKF is especially emphasized, and a covariance localization technique is needed to get EnKF working with small ensemble sizes.

We run one more experiment, where we study the effect of localization. As in the Lorenz '96 case, we apply the fifth order piecewise rational function of [19] as the tapering matrix. We choose the cutoff length experimentally for

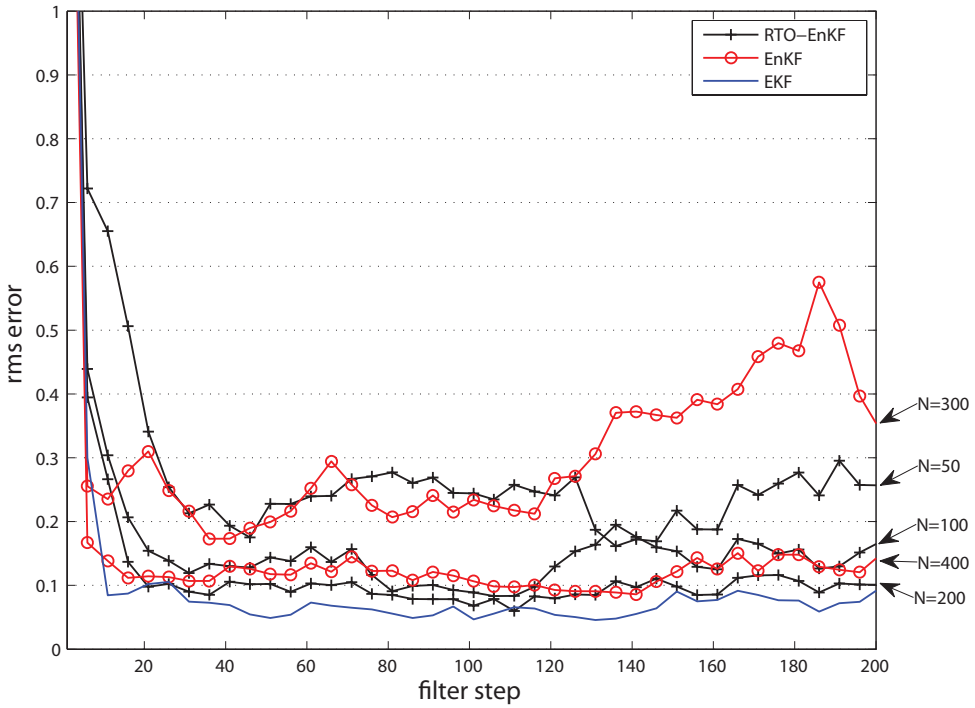

FIG. 5: Comparison of relative error of filter estimates in the QG model for EKF, EnKF with $N = 300$ and $N = 400$ and RTO-EnKF with $N = 50$, $N = 100$ and $N = 200$.

both methods so that a good filter performance is obtained. We observed that the optimally working cutoff length was different for each method; for EnKF, a smaller value gave the best performance. For RTO-EnKF, the same value did not work at all (filter diverged), and the cut-off length had to be set to a higher value to see any benefit from localization.

We compare the filter accuracy with small ensemble sizes, $N = 25$ and $N = 50$. The results are shown in Fig. 6. Again, localization significantly improves filter performance for EnKF. For $N = 25$, the localized version of EnKF yields slightly better results than RTO-EnKF run without any localization. For $N = 50$, however, the situation is the opposite: RTO-EnKF without localization is slightly more accurate than EnKF with localization. The results show that localization is less crucial in RTO-EnKF, and suggest that RTO-EnKF can work well in high-dimensional problems with small ensemble sizes without any localization applied. Note that with such small ensemble sizes, the standard EnKF without localization does not converge.

As seen from Fig. 6, the RTO-EnKF results can also be improved by including localization with a suitably chosen cutoff length; with both $N = 25$ and $N = 50$, the localized version of RTO-EnKF is more accurate than EnKF. However, the cutoff length needed here to obtain significant performance gains is rather large, and the forecast covariance matrices in RTO-EnKF are not very sparse, which means that the computationally efficient way to evaluate the cost function discussed in Section 4 and Eq. (15) is not applicable. However, as discussed earlier, relatively good performance can be obtained with RTO-EnKF without applying any localization at all.

6. CONCLUSIONS

In this paper, we present a way to apply optimization-based sampling in ensemble Kalman filtering. In the proposed method, a randomize then optimize (RTO) procedure is applied, where new ensemble members are generated by repeatedly solving an optimization problem with randomly perturbed data and prior. If the observation model is linear, the method yields independent samples from the Gaussian approximation to the posterior.

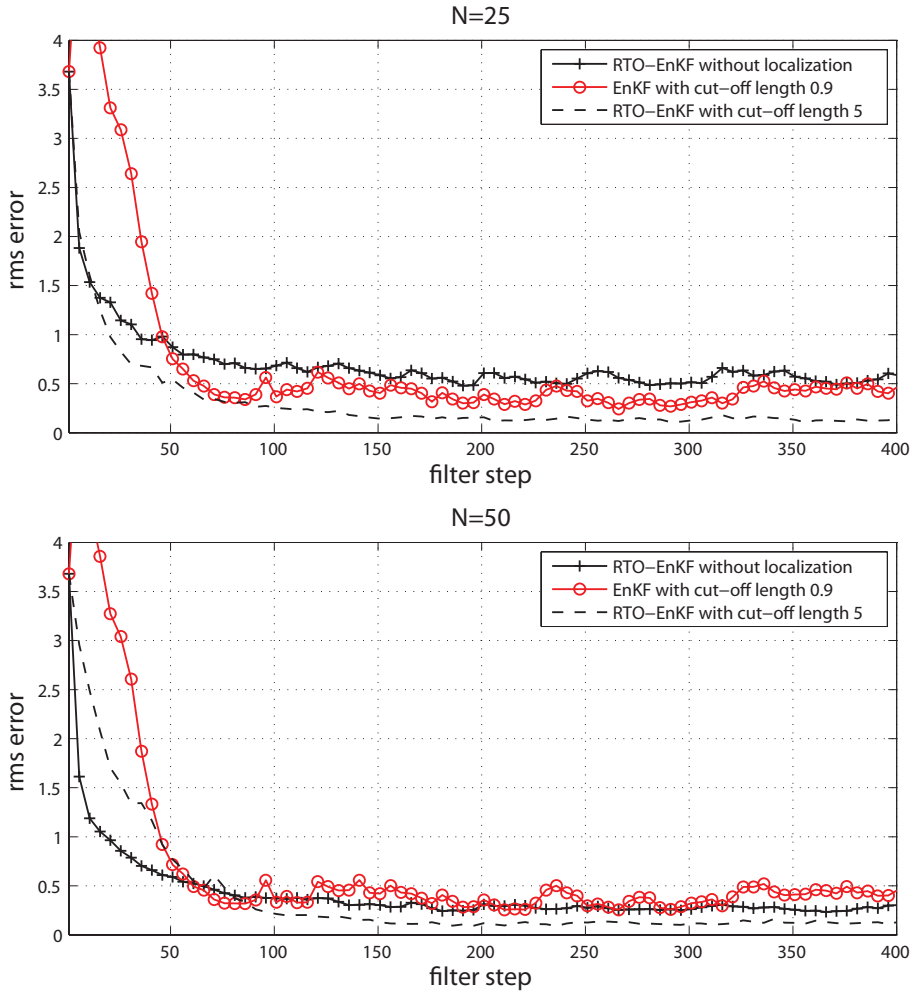

FIG. 6: Comparison of RMS errors of RTO-EnKF with and without localization and EnKF with localization for ensemble sizes $N = 25$ (top) and $N = 50$ (bottom).

In the proposed method, the prior (forecast) covariance matrix is computed as the sum of a model error covariance matrix and a sample covariance matrix computed from the propagated samples. Including the model error covariance matrix directly improves the conditioning of the forecast error covariance matrix, and diminishes the rank-deficiency problems (spurious correlations) often encountered in ensemble filtering methods. The approach has other benefits as well: for instance, including prior information of the state vector is straightforward due to the variational formulation of the method. Computationally, the approach is more demanding than many existing ensemble approaches, since it involves solving large-scale optimization problems, but the optimizations can be computed in parallel.

The proposed method, which we call RTO-EnKF, is compared to the standard ensemble Kalman filter (EnKF) in a small scale Lorenz model and a larger scale two-layer quasi-geostrophic model. In both cases, the RTO-EnKF performs better than EnKF, especially with small ensemble size when the rank-deficiency problems are pronounced.

ACKNOWLEDGMENTS

We thank Dr. Tuomo Kauranne and Prof. Heikki Järvinen for valuable comments related to the manuscript. The research is funded by the Academy of Finland.

REFERENCES

1. Kalman, R. E., A new approach to linear filtering and prediction problems, *Trans. ASME, J. Basic Eng.*, 82:35–45, 1960.
2. Lorenc, A. C., Analysis methods for numerical weather prediction, *Q. J. R. Meteorol. Soc.*, 122:1177–1194, 1986.
3. Le Dimet, F. X. and Talagrand, O., Variational algorithms for analysis and assimilation of meteorological observations: Theoretical aspects, *Tellus*, 38A:97–110, 1986.
4. Evensen, G., Sequential data assimilation with a non-linear quasi-geostrophic model using Monte Carlo methods to forecast error statistics, *J. Geophysical Res.*, 99:10143–10162, 1994.
5. Evensen, G., Sampling strategies and square root analysis schemes for the EnKF, *Ocean Dyn.*, 54:539–560, 2004.
6. Bishop, C. H., Etherton, B. J., and Majumdar, S. J.,. Adaptive sampling with the ensemble transform Kalman filter. Part I: Theoretical aspects, *Mon. Weather Rev.*, 129:420–436, 2001.
7. Anderson, J. L., An ensemble adjustment Kalman filter for data assimilation, *Mon. Weather Rev.*, 129:2884–2903, 2001.
8. Whitaker, J. S. and Hamill, T. M., Ensemble data assimilation without perturbed observations, *Mon. Weather Rev.*, 130:1913–1924, 2002.
9. Hamill, T. M., Whitaker, J. S., and Snyder, C., Distance-dependent filtering of background error covariance estimates in an ensemble Kalman filter, *Mon. Weather Rev.*, 129:2776–2790, 2001.
10. Anderson, J. L., A local least squares framework for ensemble filtering, *Mon. Weather Rev.*, 131:634–642, 2003.
11. Ott, E., Hunt, B. R., Szunyogh, I., Zimin, A. V., Kostelich, E. J., Corazza, M., Kalnay, E., Patil, D., and Yorke, J. A., A local ensemble Kalman filter for atmospheric data assimilation, *Tellus A*, 56:415–428, 2004.
12. Houtekamer, P. L. and Mitchell, H. L., Data assimilation using an ensemble Kalman filter technique, *Mon. Weather Rev.*, 126:796–811, 1998.
13. Anderson, J. L. and Anderson, S. L., A Monte Carlo implementation of the nonlinear filtering problem to produce ensemble assimilations and forecasts, *Mon. Weather Rev.*, 126:2741–2758, 1999.
14. Bardsley, J. M., MCMC-based image reconstruction with uncertainty quantification, *SIAM J. Sci. Comput.*, 34(3):A1316–A1332, 2012.
15. Bardsley, J., Solonen, A., Parker, A., Haario, H., and Howard, M., An ensemble Kalman filter using the conjugate gradient sampler, *Int. J. Uncertainty Quantification*, 3(4):357–370, 2013.
16. Solonen, A., Haario, H., Hakkarainen, J., Auvinen, H., Amour, I., and Kauranne, T., Variational ensemble Kalman filtering using limited memory BFGS, *Electronic Trans. Num. Anal.*, 39:271–285, 2012.
17. Isaksen, L., Bonavita, M., Buizza, R., Fisher, M., Haseler, J., Leutbecher, M., and Raynaud, L., Ensemble of data assimilations at ECMWF, ECMWF Technical Memorandum no. 636, 2010.
18. Furrer, R. and Bengtsson, T., Estimation of high-dimensional prior and posterior covariance matrices in Kalman filter variants, *J. Multivariate Anal.*, 98:227–255, 2007.
19. Gaspari, G. and Cohn, S. E., Construction of correlation functions in two and three dimensions, *Q. J. R. Meteorolog. Soc.*, 125:723–757, 1999.
20. Talagrand, O., Bayesian estimation. Optimal interpolation. Statistical linear estimation. In *Data Assimilation for the Earth System*, NATO Science Series, Vol. 26, pp. 21–35, Berlin, Springer, 2003.
21. Highman, N., *Accuracy and Stability of Numerical Algorithms*, 2nd ed., SIAM, Philadelphia, 2002.
22. Stroud, J. R., Stein, M. L., Lesht, B. M., Schwa, D. J., and Beletsky, D., An ensemble Kalman filter and smoother for satellite data assimilation, *J. Am. Stat. Assoc.*, 105:491, 978–990, 2010.
23. Dee, D. P., On-line estimation of error covariance parameters for atmospheric data assimilation, *Mon. Weather Rev.*, 123:1128–1145, 1995.
24. Hakkarainen, J., Ilin, A., Solonen, A., Laine, M., Haario, H., Tamminen, J., Oja, E., and Järvinen, H., On closure parameter estimation in chaotic systems, *Nonlinear Processes Geophys.*, 19:127–143, 2012.
25. Bardsley, J., Solonen, A., Haario, H., and Laine, M., Randomize-then-optimize: A method for sampling from posterior distributions in nonlinear inverse problems, *SIAM J. Sci. Comput.*, to appear, 2014.

26. Fisher, M., Leutbecher, M., and Kelly, G. A., On the equivalence between Kalman smoothing and weak-constraint four-dimensional variational data assimilation, *Q. J. R. Meteorol. Soc.*, 131(613):3235–3246. 2006.

27. Bonavita, M., Raynaud, L., and Isaksen, L., Estimating background-error variances with the ECMWF Ensemble of Data Assimilations system: Some effects of ensemble size and day-to-day variability, *Q. J. R. Meteor. Soc.*, 137B(655):423–434, 2011.

28. Lorenz, E. N., Predictability: A problem partly solved, In *Proceedings of the Seminar on Predictability*, Vol. 1, Reading, Berkshire, ECMWF, U.K., pp. 1–18, 1996.

29. Lorenz, E. N. and Emanuel, K. A., Optimal sites for supplementary weather observations: Simulation with a small model, *J. Atmos. Sci.*, 55(3):399–414, 1998.

30. Yang, S., Corazza, M., Carrassi, A., Kalnay, E., and Miyoshi T., Comparison of local ensemble transform Kalman filter, 3DVAR, and 4DVAR in a quasigeostrophic model, *Mon. Weather Rev.*, 137(2):693–709, 2009.

31. Law, K. J. H. and Stuart, A. M., Evaluating data assimilation algorithms, *Mon. Weather Rev.*, 140:3757–3782, 2012.

32. Wilks, D. S., *Statistical Methods in the Atmospheric Sciences: An Introduction*, Vol. 59, International Geophysics Series, Academic Press, Oxford, 1995.

33. Talagrand, O., Evaluation of assimilation algorithms, In *Data Assimilation: Making Sense of Observations*, Lahoz, W., Khattatov, B., and Menard, R., Eds., Springer, Germany, pp. 217–240, 2010.

34. Fandry, C. and Leslie, L., A two-layer quasi-geostrophic model of summer trough formation in the australian subtropical easterlies, *J. Atmos. Sci.*, 41:807–818, 1984.

35. Fisher, M., Tremolet, Y., Auvinen, H., Tan, D., and Poli, P., Weak-constraint and long window 4DVAR, Techn. Rep. 655, European Centre for Medium-Range Weather Forecasts, 2011.

36. Nocedal, J. and Wright, S., *Numerical Optimization*, Springer, Berlin, 1999.