

# A Moment Matching Ensemble Filter for Nonlinear Non-Gaussian Data Assimilation

JING LEI AND PETER BICKEL

*Department of Statistics, University of California, Berkeley, Berkeley, California*

(Manuscript received 1 July 2010, in final form 28 May 2011)

## ABSTRACT

The ensemble Kalman filter is now an important component of ensemble forecasting. While using the linear relationship between the observation and state variables makes it applicable for large systems, relying on linearity introduces nonnegligible bias since the true distribution will never be Gaussian. This paper analyzes the bias of the ensemble Kalman filter from a statistical perspective and proposes a debiasing method called the nonlinear ensemble adjustment filter. This new filter transforms the forecast ensemble in a statistically principled manner so that the updated ensemble has the desired mean and variance. It is also easily localizable and, hence, potentially useful for large systems. Its performance is demonstrated and compared with other Kalman filter and particle filter variants through various experiments on the Lorenz-63 and Lorenz-96 systems. The results show that the new filter is stable and accurate for challenging situations such as nonlinear, high-dimensional systems with sparse observations.

## 1. Introduction

The ensemble Kalman filter (EnKF; Evensen 1994, 2003, 2007) has become a popular tool for data assimilation because of its computational efficiency and flexibility (Houtekamer and Mitchell 1998; Anderson 2001; Whitaker and Hamill 2002; Ott et al. 2004; Evensen 2003). Although many variants have been developed, the EnKF approximates the probability distributions of the forecast state vector, the observation, and the updated state vector by Gaussian distributions. Such a Gaussian approximation allows a linear update that makes the EnKF applicable for many large-scale problems. However, in reality such a Gaussian approximation and linear update will introduce systematic bias because the true distributions can be significantly non-Gaussian and the relationship between the observation and state may be nonlinear.

A filtering approach that is adaptive to nonlinearity and non-Gaussianity is the particle filter (Gordon et al. 1993; van Leeuwen 2003). However, it is known that the ordinary particle filter requires a prohibitively large ensemble to avoid collapsing even when the dimensionality of the state space is moderately high (Snyder et al. 2008).

On the other hand, the particle filter suffers from sample impoverishment when the system is deterministic. Both the efficiency problem and sample impoverishment problem can be alleviated by reducing the dimensionality. Therefore, a major challenge for particle filters in data assimilation is localization. Traditional covariance tapering techniques for the EnKF are not applicable for the particle filter, where the forecast ensemble is updated directly from the likelihood function and the covariance matrix is not used. The sliding window localization method used in Brusdal et al. (2003) and Ott et al. (2004) seems feasible but the resampling/reweighting step of the ordinary particle filter breaks the connection between overlapping local windows. For an introduction to particle filters in data assimilation, see van Leeuwen (2009).

In this article we propose a new ensemble filtering method that enables localization by combining the advantages of both the EnKF and the particle filter. The basic idea is to view the EnKF as a linear regression of the state vector on the observation vector (Anderson 2003), where the updated state is obtained by using the true observation as predictor in the fitted linear model. Under nonlinear and non-Gaussian models, such a linear update is biased in both location (the posterior expectation) and shape (the posterior higher-order moments). The bias in location is a direct consequence of using linear regression in a nonlinear model. Our method

---

*Corresponding author address:* Jing Lei, Department of Statistics, UC Berkeley, 367 Evans Hall, Berkeley, CA 94720.  
E-mail: leij09@gmail.com

uses importance sampling to estimate the conditional expectation of the state given the observation and each ensemble member is shifted accordingly. Unlike the particle filter, which tries to estimate the whole posterior distribution, the new method uses importance sampling only to estimate the posterior mean, lending itself more easily to localization because it does not involve any reweighting or resampling steps.

Xiong et al. (2006) and Nakano et al. (2007) also use importance sampling to update the mean and covariance, and transform the forecast ensemble so that the updated ensemble has asymptotically the desired mean and covariance. In particular, Xiong et al. (2006) propose a particle filter with Gaussian resampling (PFGR) using a deterministic transform on the forecast ensemble, which is very similar to the ensemble square root filter. This method depends on the particle filter to estimate the updated mean and covariance, which is hard if the dimensionality of state space is moderately high. On the other hand, Nakano et al. (2007) develop a merging particle filter (MPF) that generates an auxiliary posterior ensemble using importance sampling and resampling, and each updated particle is obtained by a linear combination of a group of these auxiliary particles. This method gives good simulation results, but the importance sampling and resampling steps collapse frequently when the state space is high dimensional and the system is deterministic. We compare the performance of our method with both PFGR and MPF in the Lorenz-63 system. In our simulations with the Lorenz-96 system, both the PFGR and MPF degenerate. Note that in Nakano et al. (2007), the MPF does not degenerate when a stochastic Lorenz-96 system is considered.

The rest of this article is organized as follows. In section 2 we review the EnKF and the particle filter. In section 3 we examine the sources of bias in the EnKF and introduce the nonlinear ensemble adjustment filter (NLEAF). In section 4 the NLEAF algorithm is tested and compared with other methods in both the Lorenz-63 and Lorenz-96 systems. Some final remarks are given in section 5.

**2. Ensemble filters**

Filtering algorithms for data assimilation usually work sequentially. There are two major steps in each recursion. In the forecasting step, a forecast (prior) ensemble is obtained by applying the forecast model to each update (posterior) ensemble member produced at the previous time. In the update step, the forecast ensemble is modified to incorporate the information provided by the new observation. We focus on the update step, assuming that the forecast can be done in a standard way.

Formally, suppose the uncertainty of the forecast ensemble can be represented by a random variable  $\mathbf{x}$  with probability density function  $p_f(\cdot)$ , where the subindex “ $f$ ” stands for “forecast.” Assume that the observation  $\mathbf{y}$  is given by

$$\mathbf{y} = h(\mathbf{x}) + \boldsymbol{\epsilon},$$

where the observation mechanism  $h(\cdot)$  may be non-linear and  $\boldsymbol{\epsilon}$  is the observation noise independent of  $\mathbf{x}$ , with probability density function  $g(\cdot)$ , which could be non-Gaussian. Then the likelihood function of  $\mathbf{x}$  given  $\mathbf{y}$  is  $g[\mathbf{y} - h(\mathbf{x})]$ . The optimal way to update the state distribution is Bayes’s rule, in which the updated probability density function of the state variable is

$$p_a(\mathbf{x}) = \frac{p_f(\mathbf{x})g[\mathbf{y} - h(\mathbf{x})]}{\int p_f(\mathbf{x}')g[\mathbf{y} - h(\mathbf{x}')] d\mathbf{x}'},$$

where the subindex “ $a$ ” stands for “analysis.”

However, a closed form solution is available only for a few special cases such as when  $h$  is linear and  $p_f(\cdot), g(\cdot)$  are Gaussian (the Kalman filter), or when  $\mathbf{x}$  is discrete (hidden Markov models). In ensemble filtering,  $p_f$  and  $p_a$  are approximated by a discrete set (ensemble) of sample points (particles), and the ensemble is propagated by the forecast model and updated according to the observation at each time. We recall two typical ensemble filtering methods, the particle filter and the EnKF.

*a. The particle filter*

The particle filter (Gordon et al. 1993) uses importance sampling followed by a resampling step. Given the forecast ensemble  $\{\mathbf{x}_f^1, \dots, \mathbf{x}_f^n\}$  as a random sample from  $p_f$ , and the observation  $\mathbf{y}^o$ , a simple particle filter algorithm works as follows:

- 1) Evaluate the likelihood for each forecast ensemble member:  $w^j = g[\mathbf{y}^o - h(\mathbf{x}_f^j)]$  for  $j = 1, \dots, n$ .
- 2) For  $j = 1, \dots, n$ , sample the updated ensemble member  $\mathbf{x}_a^j$  independently from  $\{\mathbf{x}_f^j\}_{j=1}^n$  with weights proportional to  $(w^1, \dots, w^n)$ .

Reweighting the particles according to their likelihoods is called *importance sampling*, which dates back at least to Hammersley and Handscomb (1965). The particle filter is statistically consistent in the sense that when the ensemble size goes to infinity, the updated ensemble will be exactly a random sample from  $p_a$  (Künsch 2005). However, in high-dimensional situations, the ordinary particle filter requires a very large ensemble to search the whole state space and the ensemble tends to collapse in a few steps. Snyder et al. (2008) give a quantitative estimate of the rate of collapse in a simple Gaussian model.

Another potential problem of the particle filter is the issue of degeneracy or sample impoverishment. When resampling is used, there will inevitably be a loss of diversity of distinct particles during the update. If the system is deterministic, the updated ensemble will soon have few distinct particles. A remedy is to perturb the updated particles with small random noises, which is known as the regularized particle filter (Musso et al. 2001). Such a perturbation introduces another source of noise, which may impact performance negatively even in low-dimensional problems.

However, because of its natural advantage in dealing with nonlinear non-Gaussian problems, the particle filter is still a promising tool for data assimilation. Many resampling methods have been proposed to improve the efficiency of particle filters. Examples include, but are not limited to, Xiong et al. (2006) and Nakano et al. (2007). Of course, the major challenge is finding a good proposal density in the resampling step. The extent to which this is possible in very high-dimensional geophysical systems remains to be analyzed but see Chorin and Tu (2009) and van Leeuwen (2010) for examples in this direction.

### b. The ensemble Kalman filter

A widely used approach in data assimilation is the EnKF (Evensen 1994; Burgers et al. 1998). Unlike the particle filter, which tries to capture the whole posterior distribution, the EnKF adjusts the posterior mean and covariance to agree with the ones obtained by a Kalman filter update. Assume the observation is linear in  $\mathbf{x}$  and  $\boldsymbol{\epsilon}$ :

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \boldsymbol{\epsilon},$$

with Gaussian noise  $\boldsymbol{\epsilon} \sim \text{Norm}(0, \mathbf{R})$ . Given the forecast ensemble  $\{\mathbf{x}_f^1, \dots, \mathbf{x}_f^n\}$  and true observation  $\mathbf{y}^o$ , the stochastic EnKF works as follows:

- 1) Compute the forecast sample covariance matrix:  $\mathbf{P}_f = n^{-1} \sum_{j=1}^n (\mathbf{x}_f^j - \bar{\mathbf{x}}_f)(\mathbf{x}_f^j - \bar{\mathbf{x}}_f)^T$ , where  $\bar{\mathbf{x}}_f = n^{-1} \sum_{j=1}^n \mathbf{x}_f^j$  is the forecast ensemble mean, and the superscript T means matrix transpose.
- 2) Generate perturbed observations:  $\mathbf{y}^j = \mathbf{H}\mathbf{x}_f^j + \boldsymbol{\epsilon}^j$ , with  $\boldsymbol{\epsilon}^j \sim \text{Norm}(0, \mathbf{R})$ .
- 3) Estimate the linear regression coefficient of  $\mathbf{x}$  on  $\mathbf{y}$ :  $\mathbf{K} = \mathbf{P}_f \mathbf{H}^T (\mathbf{H} \mathbf{P}_f \mathbf{H}^T + \mathbf{R})^{-1}$ . This is usually done by multiplying the sample covariance between the forecast ensemble and the perturbed observations and the inverse of the sample variance of the perturbed observations.
- 4) Update ensemble:  $\mathbf{x}_a^j = \mathbf{x}_f^j + \mathbf{K}(\mathbf{y}^o - \mathbf{y}^j)$ .

It is known that using perturbed observations will introduce some sampling bias especially for small ensembles. Some systematic sampling schemes are often

used to make sure that the perturbations have zero mean, covariance  $\mathbf{R}$ , and zero correlation with the forecast ensemble (Pham 2001). Another technique known as covariance inflation is often used to mitigate the forecast ensemble bias carried over from the previous time. There is vast literature on different variants and applications of the EnKF. See Evensen (2003) for a nice review and Mitchell and Houtekamer (2009) for more recent developments.

Some ensemble filters update the first two moments of the forecast ensemble in a deterministic manner, under the name of Kalman square root filter (Anderson 2001; Bishop et al. 2001; Whitaker and Hamill 2002). Although deterministic filters have no sampling error introduced by perturbing the observations, the bias caused by the nonlinearity and non-Gaussianity still remains even when the ensemble is large. There have been many studies comparing the stochastic and deterministic filters. It is noted under some simple models, for example in Lawson and Hansen (2004) and Lei et al. (2010), that the stochastic filter is more robust against nonlinearity and non-Gaussianity.

A key feature of the EnKF is that it updates each particle directly, without reweighting or resampling, which makes it easily localizable. In the next section we will introduce another filter with the same property but having reduced bias in a nonlinear, non-Gaussian context.

## 3. A moment-matching nonlinear non-Gaussian ensemble filter

### a. Why does the ensemble Kalman filter work?

An explanation of the stochastic EnKF is based on the following simple fact: if  $\mathbf{x}$  and  $\mathbf{y}$  are jointly Gaussian, then the posterior distribution of  $\mathbf{x}$  given  $\mathbf{y}$  depends on  $\mathbf{y}$  only through the mean. That is, for any  $\mathbf{y}$ , let  $\boldsymbol{\mu}_a(\mathbf{y})$  be the posterior mean of  $\mathbf{x}$  given  $\mathbf{y}$ , then the posterior distribution of  $\mathbf{x}$  given  $\mathbf{y}$  is  $\text{Norm}[\boldsymbol{\mu}_a(\mathbf{y}), \mathbf{P}_a]$ , where  $\mathbf{P}_a$  does not depend on  $\mathbf{y}$ . By the construction of the perturbed observation  $\mathbf{y}^j$ ,  $(\mathbf{x}_f^j, \mathbf{y}^j)$  is jointly Gaussian. Therefore, recalling that  $\boldsymbol{\mu}_f$  is the forecast ensemble mean,  $\mathbf{x}_f^j - \boldsymbol{\mu}_a(\mathbf{y}^j) = (\mathbf{I} - \mathbf{K}\mathbf{H})(\mathbf{x}_f^j - \boldsymbol{\mu}_f) + \mathbf{K}\boldsymbol{\epsilon}^j$  is a random draw from  $\text{Norm}(0, \mathbf{P}_a)$  and  $\mathbf{x}_a^j = \mathbf{x}_f^j - \boldsymbol{\mu}_a(\mathbf{y}^j) + \boldsymbol{\mu}_a(\mathbf{y}^o)$  is a random draw from  $\text{Norm}[\boldsymbol{\mu}_a(\mathbf{y}^o), \mathbf{P}_a]$ , the true posterior distribution. On the other hand, joint Gaussianity also implies that  $\boldsymbol{\mu}_a(\mathbf{y})$  is a linear function of  $\mathbf{y}$  with coefficient  $\mathbf{K}$  as given in step 2 in section 2b. Thus the update can be written as  $\mathbf{x}_a^j = \mathbf{x}_f^j + \mathbf{K}(\mathbf{y}^o - \mathbf{y}^j)$ . Note that by definition we have  $\mathbf{y}^j = \mathbf{H}\mathbf{x}_f^j + \boldsymbol{\epsilon}^j$ . As a result, the update formula can be written as  $\mathbf{x}_a^j = \mathbf{x}_f^j + \mathbf{K}[(\mathbf{y}^o - \boldsymbol{\epsilon}^j) - \mathbf{H}\mathbf{x}_f^j]$ , which is exactly the same as the ‘‘observation perturbation’’ used in Eq. (13) of Burgers et al. (1998). In this paper we use the perturbed observation  $\mathbf{y}^j$  to present

a unified argument for both Gaussian linear and non-Gaussian nonlinear models.

In a non-Gaussian nonlinear model, the EnKF will be biased for two reasons. First,  $\mathbf{K}(\mathbf{y}^o - \mathbf{y}^j) \neq \boldsymbol{\mu}_a(\mathbf{y}^o) - \boldsymbol{\mu}_a(\mathbf{y}^j)$  since  $\boldsymbol{\mu}_a(\mathbf{y})$ , the conditional expectation of  $\mathbf{x}$  given  $\mathbf{y}$ , is no longer linear in  $\mathbf{y}$ . Second, the recentered conditional random variables  $\mathbf{x}_f^j - \boldsymbol{\mu}_a(\mathbf{y}^j) + \boldsymbol{\mu}_a(\mathbf{y}^o)$  will not have the correct variance and higher moments. The first source of bias will result in a larger distance between the true state and the updated ensemble mean, whereas the second source of bias will affect the shape of the updated ensemble. Such a shape bias might lead to further location bias when propagated by the dynamics.

*b. The NLEAF algorithm with first-order correction*

We now introduce the NLEAF as a debiasing alternative to the EnKF. It requires no parametric assumptions on the prior distribution of  $\mathbf{x}$ , observation function  $h(\cdot)$ , and observation noise distribution  $g(\cdot)$ . The basic idea is to estimate  $\boldsymbol{\mu}_a(\mathbf{y})$  using importance sampling, instead of linear fitting as in the EnKF. Let  $p_f(\cdot)$  and  $p_a(\cdot)$  be the forecast (prior) and updated (posterior given  $\mathbf{y}$ ) densities, respectively. Then we have, for any  $\mathbf{y}$ ,

$$p_a(\mathbf{x}) = \frac{p_f(\mathbf{x})g[\mathbf{y} - h(\mathbf{x})]}{\int p_f(\mathbf{x}')g[\mathbf{y} - h(\mathbf{x}')] d\mathbf{x}'}$$

$$\Rightarrow \boldsymbol{\mu}_a(\mathbf{y}) = \frac{\int \mathbf{x}p_f(\mathbf{x})g[\mathbf{y} - h(\mathbf{x})] d\mathbf{x}}{\int p_f(\mathbf{x})g[\mathbf{y} - h(\mathbf{x})] d\mathbf{x}}. \tag{1}$$

The importance sampling estimator of the conditional expectation is given by

$$\hat{\boldsymbol{\mu}}_a(\mathbf{y}) = \frac{\sum_{j=1}^n \mathbf{x}_f^j g[\mathbf{y} - h(\mathbf{x}_f^j)]}{\sum_{j=1}^n g[\mathbf{y} - h(\mathbf{x}_f^j)]}. \tag{2}$$

Given the forecast ensemble  $\{\mathbf{x}_f^1, \dots, \mathbf{x}_f^n\}$  and observation  $\mathbf{y}^o$ , the NLEAF update works as follows:

- 1) Generate perturbed observations  $\mathbf{y}^j = h(\mathbf{x}_f^j) + \boldsymbol{\epsilon}^j$ , with  $\boldsymbol{\epsilon}^j$  independently sampled from probability density function  $g(\cdot)$ .
- 2) Estimate the conditional expectation  $\hat{\boldsymbol{\mu}}_a(\mathbf{y})$  using (2) for  $\mathbf{y}$  equal to the true observation  $\mathbf{y}^o$  and all  $\mathbf{y}^j, j = 1, \dots, n$ .
- 3) Update ensemble:  $\mathbf{x}_a^j = \mathbf{x}_f^j + \hat{\boldsymbol{\mu}}_a(\mathbf{y}^o) - \hat{\boldsymbol{\mu}}_a(\mathbf{y}^j)$ .

This algorithm improves the EnKF by using importance sampling to estimate the conditional expectations. It is

known that under mild conditions the importance sampling estimator  $\hat{\boldsymbol{\mu}}_a(\mathbf{y})$  converges to  $\boldsymbol{\mu}_a(\mathbf{y})$  as the ensemble size tends to infinity (Künsch 2005). Therefore  $\mathbf{x}_a^j$  is centered approximately at  $\boldsymbol{\mu}_a(\mathbf{y}^o)$ , since  $\mathbf{x}_f^j$  is centered approximately at  $\boldsymbol{\mu}_a(\mathbf{y}^j)$  conditioning on  $\mathbf{y}^j$ . As a result, the first source of bias in the EnKF is reduced. On the other hand, it also keeps the simplicity of the EnKF, avoiding the reweighting and resampling steps used by the particle filter. This is particularly suitable for sliding-window localization techniques (Brusdal et al. 2003; Ott et al. 2004), because when applied to localized state vectors, the absence of resampling and reweighting maintains the spatial smoothness across neighboring local state vectors. As a result, one would expect the NLEAF algorithm to be applicable for many high-dimensional problems with a better accuracy than the EnKF. Before demonstrating its performance in section 4, we describe two important extensions of the NLEAF.

*c. Higher-order corrections*

The NLEAF update described above only adjusts the posterior mean. In fact the same idea can be applied for higher-order corrections. For example, an NLEAF with second-order correction uses importance sampling to estimate the posterior covariance of  $\mathbf{x}$  given  $\mathbf{y}$ , denoted as  $\mathbf{P}_a(\mathbf{y})$ . Theoretically we have

$$\mathbf{P}_a(\mathbf{y}) = \int [\mathbf{x} - \boldsymbol{\mu}_a(\mathbf{y})][\mathbf{x} - \boldsymbol{\mu}_a(\mathbf{y})]^T p_a(\mathbf{x}) d\mathbf{x},$$

where  $p_a(\mathbf{x})$  is defined as in (1). Then the importance sampling estimate for  $\mathbf{P}_a(\mathbf{y})$  is

$$\hat{\mathbf{P}}_a(\mathbf{y}) = \frac{\sum_{j=1}^n [\mathbf{x}_f^j - \hat{\boldsymbol{\mu}}_a(\mathbf{y})][\mathbf{x}_f^j - \hat{\boldsymbol{\mu}}_a(\mathbf{y})]^T g[\mathbf{y} - h(\mathbf{x}_f^j)]}{\sum_{j=1}^n g[\mathbf{y} - h(\mathbf{x}_f^j)]}, \tag{3}$$

where  $\hat{\boldsymbol{\mu}}_a(\mathbf{y})$  is obtained using (2). The update with second-order correction is

$$\mathbf{x}_a^j = \hat{\boldsymbol{\mu}}_a(\mathbf{y}^o) + \hat{\mathbf{P}}_a(\mathbf{y}^o)^{1/2} \hat{\mathbf{P}}_a(\mathbf{y}^j)^{-1/2} [\mathbf{x}_f^j - \hat{\boldsymbol{\mu}}_a(\mathbf{y}^j)].$$

To understand this algorithm, note that when the ensemble size is large,  $\hat{\boldsymbol{\mu}}_a(\mathbf{y}) \approx \boldsymbol{\mu}_a(\mathbf{y})$  and  $\hat{\mathbf{P}}_a(\mathbf{y}) \approx \mathbf{P}_a(\mathbf{y})$  for all  $\mathbf{y}$ , then  $\mathbf{x}_f^j - \boldsymbol{\mu}_a(\mathbf{y}^j)$  has mean zero and covariance  $\mathbf{P}_a(\mathbf{y}^j)$ . Therefore the updated ensemble member  $\mathbf{x}_a^j$  has approximately mean  $\boldsymbol{\mu}_a(\mathbf{y}^o)$  and covariance  $\mathbf{P}_a(\mathbf{y}^o)$ . As shown in our numerical experiments, this higher-order correction does improve the accuracy. It even outperforms the particle filter in some settings with deterministic dynamics. As a trade-off, it requires a larger ensemble size

and is computationally more intensive than the first-order NLEAF because of the covariance matrix estimation and inversion. We also note this is actually a stochastic version of the Gaussian resampling particle filter proposed by Xiong et al. (2006).

#### d. NLEAF with unknown likelihood function

Another useful extension of the NLEAF is to use nonlinear regression methods to estimate  $\mu_a(\mathbf{y})$ . When  $\mathbf{y}$  is generated by a black-box function, the relationship between the observation  $\mathbf{y}$  given the state vector  $\mathbf{x}$  might be complicated so that the density function of  $\mathbf{y}$  given  $\mathbf{x}$  is not available in analytic form. One practical example is satellite radiance. Satellite radiance is related to temperature, humidity, and other trace variables depending on the wavelength. But no simple analytic relationship is known between the state and the observation. In this case, all we have is the forecast ensemble paired with the perturbed observations:  $\{(\mathbf{x}_f^j, \mathbf{y}^j)\}_{j=1}^n$ . One may use regression methods to estimate the conditional moments of  $\mathbf{x}$  given  $\mathbf{y}$ . For example, the EnKF uses a linear regression to estimate  $\mu_a(\mathbf{y})$ . One can also use more general methods, such as polynomial regression, to handle nonlinearity. More specifically, one may estimate  $\mu_a(\mathbf{y})$  as a function of  $\mathbf{y}$  by minimizing over all quadratic functions  $m(\cdot)$  under the square loss  $\sum_{j=1}^n \|m(\mathbf{y}^j) - \mathbf{x}_f^j\|_2^2$ . This could be a promising method for nonlinear data assimilation problems where we lack knowledge of the observation-generating mechanism.

## 4. Numerical experiments

### a. The Lorenz-63 system

The Lorenz-63 system is a three-dimensional model determined by an ordinary differential equation system:

$$d\mathbf{z}_\tau(1)/d\tau = -\sigma\mathbf{z}_\tau(1) + \sigma\mathbf{z}_\tau(2), \quad (4)$$

$$d\mathbf{z}_\tau(2)/d\tau = -\mathbf{z}_\tau(1)\mathbf{z}_\tau(3) + \rho\mathbf{z}_\tau(1) - \mathbf{z}_\tau(2), \quad (5)$$

$$d\mathbf{z}_\tau(3)/d\tau = \mathbf{z}_\tau(1)\mathbf{z}_\tau(2) - \beta\mathbf{z}_\tau(3), \quad (6)$$

where  $\mathbf{z}_\tau$  is the three-dimensional state vector describing a simplified flow of heated fluid with  $\tau$  being the continuous time index. The parameters are set as  $\beta = 8/3$ ,  $\rho = 28$ , and  $\sigma = 10$ .

In our simulation the system is discretized using the fourth-order Runge–Kutta method. Let  $\mathbf{x}_t = \mathbf{z}_{\Delta t}$ , for all integers  $t$ , with  $\Delta$  being the step size. A larger value of  $\Delta$  indicates a more nonlinear relationship between  $\mathbf{x}_t$  and  $\mathbf{x}_{t+1}$ . For a given starting state  $\mathbf{x}_0^*$ , a hidden true orbit  $\{\mathbf{x}_t^*, t \geq 0\}$  and observation  $\mathbf{y}_t^o = \mathbf{x}_t^* + \boldsymbol{\epsilon}_t$  are generated using independent observation noise sequence  $\boldsymbol{\epsilon}_t$  with

mean 0 and covariance  $\theta^2\mathbf{I}$ . Starting from an initial ensemble  $\{\mathbf{x}_0^j\}_{j=1}^n$ , filtering methods are applied recursively at each time  $t \geq 1$ . The updated ensemble average is used as the single point estimate of  $\mathbf{x}_t^*$ . The main evaluation criterion used is the root-mean-squared error (RMSE):  $\|\bar{\mathbf{x}}_{t,a} - \mathbf{x}_t^*\|_2/\sqrt{d}$ , where  $\|\cdot\|_2$  is the Euclidean norm,  $d = 3$  is the dimensionality of the state space, and  $\bar{\mathbf{x}}_{t,a}$  is the updated ensemble mean at time  $t$ . For a measurement of the ensemble spread (sharpness) we look at  $[\text{trace}(\hat{\mathbf{P}}_{t,a})/d]^{1/2}$  in comparison with the RMSE, where  $\hat{\mathbf{P}}_{t,a}$  is the sample covariance matrix of the updated ensemble at time  $t$ . In addition, we use the percentage of  $\mathbf{x}_t^*$  being covered by the range between 0.025 and 0.975 quantiles (the sample 95% confidence interval) of the updated ensemble as a measurement of how well the updated ensemble covers the true state.

### 1) SIMULATION SETUP

We look at two different observation error distributions. First we consider Gaussian observation noise, which has been used widely in previous studies. In this case, we let the coordinates of  $\boldsymbol{\epsilon}_t$  be independent Gaussian with mean 0 and variance  $\theta^2$ . On the other hand, in order to study non-Gaussian observation noises we let  $\boldsymbol{\epsilon}_t$  have independent coordinates with a common double-exponential distribution (also known as the Laplace distribution) with mean 0 and variance  $2\theta^2$ . The density of such a double-exponential distribution is  $p(z) = (2\theta)^{-1} \exp(-|z|/\theta)$ . The EnKF variants use the noise variance  $2\theta^2$  in the update and pretend that the noise is still Gaussian. The NLEAF update uses the true double-exponential density as the likelihood.

The initial ensemble is generated by applying the standard EnKF with 400 ensemble members and standard Gaussian observation noise after 10 000 assimilation cycles. At each following time step, for each filtering method in comparison, a forecast ensemble of size  $n = 400$  is obtained by applying the fourth-order Runge–Kutta method on the corresponding previous update ensemble. A noisy observation of the true state is also generated. Then each ensemble filtering method is applied individually to the corresponding forecast ensemble and the observation, obtaining the new update ensemble. The number of assimilation cycles is chosen to be 2000 throughout our presentation to make the results quickly reproducible. For both the Lorenz-63 system and the Lorenz-96 system (see section 4b), the same results are obtained in simulations with 50 000 assimilation cycles. We consider two different step sizes (denoted  $\Delta$ ), 0.02 and 0.05, to represent different levels of nonlinearity. Moreover, three values of the noise scale parameter  $\theta$  are considered: 0.5, 1, and 2.

TABLE 1. Average RMSE over 2000 time steps for the Lorenz-63 system with Gaussian  $N(0, \theta^2)$  observation errors and step size  $\Delta$ . Also presented are the average ensemble spread (in parentheses) and the percentage of  $\mathbf{x}_t^*$  (3) being ranked between 2.5% and 97.5% in the updated ensemble (below the average RMSE and standard deviation).

$\theta$	$\Delta = 0.02$			$\Delta = 0.05$		
	1/2	1	2	1/2	1	2
EnKF	0.037 (0.047) 94.9	0.077 (0.102) 96.3	0.163 (0.239) 97.1	0.058 (0.073) 95.7	0.137 (0.165) 94.7	0.368 (0.412) 90.3
NLEAF1	0.038 (0.047) 95.2	0.079 (0.101) 95.1	0.171 (0.226) 95.9	0.057 (0.070) 96.0	0.132 (0.158) 95.2	0.320 (0.365) 94.8
NLEAF2	0.034 (0.041) 95.2	0.066 (0.083) 95.4	0.130 (0.172) 95.9	0.049 (0.057) 95.5	0.098 (0.118) 96.6	0.212 (0.264) 95.6
PF	0.049 (0.076) 98.6	0.088 (0.141) 98.7	0.177 (0.278) 98.3	0.062 (0.087) 98.3	0.116 (0.160) 98.2	0.232 (0.326) 98.3
PFGR	0.039 (0.059) 96.1	0.078 (0.122) 97.6	0.178 (0.251) 96.3	0.060 (0.076) 96.1	0.116 (0.159) 97.1	0.279 (0.345) 94.0
MPF	0.039 (0.061) 98.6	0.081 (0.126) 98.2	0.175 (0.258) 97.2	0.056 (0.074) 97.5	0.117 (0.157) 97.1	0.274 (0.351) 96.7

The methods in comparison are: the stochastic EnKF (EnKF), NLEAF with first-order correction (NLEAF1), and NLEAF with second-order correction (NLEAF2), particle filter (PF), PFGR (Xiong et al. 2006), and MPF (Nakano et al. 2007). PFGR and MPF use inflation

$$\mathbf{x}_a^j + (1 + \delta)(\mathbf{x}_a^j - \bar{\mathbf{x}}_a), \tag{7}$$

with  $\delta = 0.01$  to avoid degeneracy. The PF uses a slightly different inflation method that replaces the  $j$ th updated ensemble member  $\mathbf{x}_a^j$  by  $\mathbf{x}_a^j + 2\delta \text{Cov}(\mathbf{x}_a)^{1/2} \boldsymbol{\xi}^j$ , where  $\text{Cov}(\mathbf{x}_a) = \langle \mathbf{x}_a^j \mathbf{x}_a^{jT} \rangle - \langle \mathbf{x}_a^j \rangle \langle \mathbf{x}_a^j \rangle^T$  is the posterior covariance matrix and  $\boldsymbol{\xi}^j$  is an independent Gaussian noise with unit variance.

*Remark 1:* The EnKF used in all of the simulations is the standard single ensemble EnKF, which typically needs inflation in nonlinear, high-dimensional applications.

Given the large ensemble size used here, no covariance inflation is used with the EnKF in the Lorenz-63 experiments.

## 2) RESULTS

The simulation results are summarized in Table 1 and Table 2. The first number in each cell is the average RMSE over 2000 steps. The number in the parentheses is the average ensemble spread measured by the square root of trace of the ensemble covariance matrix, which is scaled by  $1/\sqrt{3}$  to match the RMSE. The number below the RMSE and ensemble spread is the percentage of  $\mathbf{x}_t^*$  (3) being covered by the 95% confidence interval.

A good filtering method should produce 1) a small average RMSE and 2) an ensemble spread that well represents the ensemble mean error (Sacher and Bartello 2008). If the spread is too small, the updated ensemble is

TABLE 2. Average RMSE over 2000 time steps for the Lorenz-63 system with step size  $\Delta$ . The observation errors are sampled independently from a double-exponential distribution with mean zero and scale parameter  $\theta$ . Also presented is the average ensemble spread (in parentheses) and the percentage of  $\mathbf{x}_t^*$  (3) being ranked between 2.5% and 97.5% in the updated ensemble (below the average RMSE and standard deviation).

$\theta$	$\Delta = 0.02$			$\Delta = 0.05$		
	1/2	1	2	1/2	1	2
EnKF	0.050 (0.070) 97.4	0.109 (0.156) 96.4	0.279 (0.388) 94.6	0.092 (0.109) 95.4	0.223 (0.263) 94.2	0.524 (0.618) 94.6
NLEAF1	0.044 (0.054) 95.5	0.093 (0.118) 95.0	0.212 (0.267) 94.5	0.081 (0.088) 93.1	0.176 (0.198) 93.9	0.432 (0.466) 92.4
NLEAF2	0.043 (0.046) 90.5	0.079 (0.096) 93.9	0.156 (0.201) 96.2	0.060 (0.070) 95.3	0.129 (0.147) 94.1	0.295 (0.324) 92.7
PF	0.057 (0.081) 97.8	0.104 (0.154) 97.4	0.204 (0.296) 97.8	0.073 (0.095) 98.2	0.138 (0.180) 97.9	0.298 (0.366) 95.5
PFGR	0.052 (0.068) 94.9	0.103 (0.139) 96.6	0.209 (0.287) 96.7	0.074 (0.089) 95.5	0.167 (0.185) 93.8	0.343 (0.417) 95.2
MPF	0.052 (0.071) 96.5	0.096 (0.140) 97.6	0.207 (0.288) 97.2	0.071 (0.086) 95.2	0.148 (0.189) 98.0	0.338 (0.410) 94.6

not likely to cover the true state; and if the spread is too large, the updated ensemble would be too non-informative. In other words, the true state should look like a random sample from the updated ensemble. Therefore, the ensemble confidence interval coverage of the true state would be a good indicator of the filter performance. Based on our experiment [see also Figs. 1–6 in Sacher and Bartello (2008) and discussions in Mitchell and Houtekamer (2009)], it is usually good to have the updated ensemble spread close to or slightly larger than the RMSE.

As we see from the tables, the NLEAF gives unparalleled performance except for the double-exponential noise with  $\Delta = 0.05$  and  $\theta = 2$ , which is the hardest case with a significantly nonlinear system and large observation noise. Even in this case, it produces similar average RMSE as the particle filter, with a tighter ensemble spread and a reasonable confidence interval coverage.

The EnKF gives large RMSEs because of its bias under nonlinearity and non-Gaussianity. As for the particle filter, in section 2a we point out that it has the issue of sample impoverishment in deterministic systems. In our implementation, small random perturbations are added to the updated ensemble members to overcome this difficulty, increasing both the uncertainty of the update ensemble and the confidence interval coverage. It is interesting to compare the performance of the PF to NLEAF1. When  $\Delta = 0.05$ , NLEAF1 gives slightly larger RMSE. Conversely, NLEAF1 performs better than the PF when  $\Delta = 0.02$ . This is because when the step size is small, the system is more linear and ignoring higher-order moments does not lose much information.

We observe that for double-exponential observations, NLEAF2 tends to produce a tight ensemble, which might fail to cover the true state. In practice one can mitigate this issue by inflating the updated ensemble covariance as in (7) with some small positive constant  $\delta$ .

### b. The Lorenz-96 system

The Lorenz-96 system (Lorenz 1996) is another common test bed for data assimilation algorithms (Bengtsson et al. 2003; Ott et al. 2004; Anderson 2007). The state vector has 40 coordinates evolving according to the following ordinary differential equation system:

$$\begin{aligned} d\mathbf{z}_\tau(i)/d\tau = & [\mathbf{z}_\tau(i+1) - \mathbf{z}_\tau(i-2)]\mathbf{z}_\tau(i-1) \\ & - \mathbf{z}_\tau(i) + 8, \quad \text{for } i = 1, \dots, 40, \end{aligned} \quad (8)$$

where  $\mathbf{z}_\tau(0) = \mathbf{z}_\tau(40)$ ,  $\mathbf{z}_\tau(-1) = \mathbf{z}_\tau(39)$  and  $\mathbf{z}_\tau(41) = \mathbf{z}_\tau(1)$ . This system mimics the evolution of some meteorological quantity at 40 equally spaced grid points along

a latitude circle. The attractor has 13 positive local Lyapunov vectors. Similarly, the system is discretized with step size  $\Delta$ :  $\mathbf{x}_t = \mathbf{z}_{\Delta t}$ , for all integers  $t$ .

In such a high-dimensional problem, the particle filter often collapses in a single step with all weight concentrating on one particle. The EnKF also has difficulties when the sample size  $n$  is as small as a few tens if no dimension-reduction techniques are used. Most dimension-reduction techniques for the Lorenz-96 system are based on the assumption that two coordinates of  $\mathbf{x}_t$  have little dependence if they are far away in physical space. For example, the correlation between  $\mathbf{x}(1)$  and  $\mathbf{x}(2)$  might be significant, but  $\mathbf{x}(1)$  and  $\mathbf{x}(20)$  would be nearly independent. It should be noted that in the Lorenz-96 model described in (8), the correlation between  $\mathbf{x}(j)$  and  $\mathbf{x}(j+1)$  is statistically detectable but rather weak [see Lorenz (2005), which also introduces models with stronger spatial correlation]. Based on such a spatial correlation assumption, the covariance tapering idea (Gaspari and Cohn 1999; Houtekamer and Mitchell 2001) is applicable. Houtekamer and Mitchell (1998) used a localization scheme in which only data points within a cutoff horizontal distance will be used for the update of each horizontal point. The NLEAF implementation uses the sliding-window localization (Brusdal et al. 2003; Ott et al. 2004), where local observations are used to update local state vectors, and the whole state vector is reconstructed by aggregating the local updates. Specifically, the state vector  $\mathbf{x} = [\mathbf{x}(1), \dots, \mathbf{x}(40)]$  is broken down to overlapping local vectors  $\mathbf{x}(\mathbf{N}_j)$ ,  $j = 1, \dots, 40$ , with  $\mathbf{N}_j = (j-l, \dots, j, \dots, j+l)$ , for some positive integer  $l$  and all numbers being mod 40. Then each local vector  $\mathbf{x}(\mathbf{N}_j)$  is updated by the ensemble filter using local observations  $\mathbf{y}(\mathbf{N}_j)$  (assuming  $\mathbf{y} = \mathbf{x} + \boldsymbol{\epsilon}$ ). Therefore each coordinate  $\mathbf{x}(j)$  is updated simultaneously in  $2l+1$  local vectors including  $\mathbf{x}(\mathbf{N}_{j-l}), \mathbf{x}(\mathbf{N}_{j-l+1}), \dots, \mathbf{x}(\mathbf{N}_{j+l})$ . The final update for  $\mathbf{x}(j)$  is then obtained by averaging its updates in local vectors  $\mathbf{x}(\mathbf{N}_{j-k}), \mathbf{x}(\mathbf{N}_{j-k+1}), \dots, \mathbf{x}(\mathbf{N}_{j+k})$  for some positive integer  $k \leq l$ .

Our study of the Lorenz-96 system consists of two different settings corresponding to two different levels of difficulty.

#### 1) THE HARD CASE

##### (i) Simulation setup

This setting has been considered by Bengtsson et al. (2003) as an early effort toward a high-dimensional nonlinear non-Gaussian filter. The setup is similar to previous simulations in the Lorenz-63 system. The initial ensemble is generated by applying the EnKF of size 400 after 2000 steps using a complete observation model

with standard Gaussian noise. In the next 2000 assimilation cycles, different filtering methods are applied with an ensemble of size 400. The time step between two assimilation cycles is  $\Delta = 0.4$ , which indicates a highly nonlinear non-Gaussian forecast ensemble. To make the problem more challenging, it is assumed that not all coordinates of the state variable are observed. In particular, a linear but incomplete observation mechanism is used:  $\mathbf{y}_t = \mathbf{H}\mathbf{x}_t + \boldsymbol{\epsilon}_t$ , where  $\mathbf{H} = (\mathbf{e}_1, \mathbf{e}_3, \dots, \mathbf{e}_{39})^T$  with  $\mathbf{e}_i = (0, \dots, 1, \dots, 0)^T$  being the unit vector with all zero entries except the  $i$ th position, and  $\boldsymbol{\epsilon}_t \sim \text{Norm}(0, \mathbf{I}_{20}/2)$ . That is, only state coordinates with odd indices are observed with independent Gaussian noise of variance 0.5. Such a combination of incomplete observation, high nonlinearity, and high dimensionality poses great challenge to data assimilation algorithms.

Here we compare four methods: 1) NLEAF1 (NLEAF with first-order correction); 2) NLEAF1q (NLEAF using quadratic regression); 3) EnKF (the stochastic EnKF); and 4) XEnsF, a nonlinear filter using Gaussian mixture approximations to the prior and posterior distributions (Bengtsson et al. 2003). Both NLEAF1 and NLEAF1q uses localization and averaging parameters  $(l, k) = (3, 1)$ . The inflation follows (7) with  $\delta = 0.045$  for NLEAF1 and NLEAF1q, and  $\delta = 0.005$  for the EnKF.

(ii) Results

The results are summarized in Table 3. We only look at the mean, median, and standard deviation of the 2000 RMSEs since they are the only reported performance metrics for the XEnsF in Bengtsson et al. (2003). Both the NLEAF1 and NLEAF1q give small and stable RMSEs on average. Remember that in this setting the observation is still linear with Gaussian noise, the NLEAF algorithms outperform other methods because of their ability of handling nonlinearity. One might note that the difference in the average RMSE seems not too large compared to the standard deviation. The reason is that the 2000 RMSEs have strong temporal correlation and their distributions are heavy tailed.

It should be noted that the result for XEnsF is quoted directly from Bengtsson et al. (2003), thus their results depend on different simulation data than ours and are perhaps less comparable.

2) THE EASY CASE

(i) Simulation setup

The easy case uses observation interval  $\Delta = 0.05$ , which makes the system much more linear and can be thought nominally equivalent to 6 h in real-world time (Lorenz 1996). This setting allows a more comprehensive comparison with existing methods since it has been studied

TABLE 3. Average RMSE over 2000 time steps for the Lorenz-96 system with incomplete observations and Gaussian  $N(0, 0.5)$  observation errors. The step size  $\Delta = 0.4$ .

	NLEAF1	NLEAF1q	EnKF	XEnsF
Mean	0.68	0.74	0.79	0.92
Median	0.63	0.67	0.74	0.85
Std dev	0.27	0.28	0.29	0.31

extensively in the literature (e.g., see Whitaker and Hamill 2002; Ott et al. 2004; Sakov and Oke 2008a,b). In particular, we can study the effect of the ensemble size  $n$  on filter performance, which is an important practical concern.

The initial ensemble is generated using the same method as in the hard case. Then the system runs to generate 2000 assimilation cycles with a complete observation  $\mathbf{y}_t = \mathbf{x}_t + \boldsymbol{\epsilon}_t$ . Since we are interested in non-Gaussian situations, the observation error  $\boldsymbol{\epsilon}_t$  has independent double-exponential coordinates with scale parameter  $\theta = 1$ .

We compare the NLEAF with first-order correction (NLEAF1), the EnKF, and the local ensemble transform Kalman filter (LETKF; Ott et al. 2004). The LETKF is among the best-performing ensemble filters in this setting with Gaussian observation noise. It uses the sliding-window localization and performs well especially when the ensemble size is very small. However, it is also reported that its performance can hardly be improved by a larger ensemble size. Other particle filter-based methods such as the PFGR and MPF either have difficulty in localization or diverge with limited ensemble size for deterministic systems.

The LETKF uses localization parameter  $k = l = 3$  for  $n = 10, 20$ ,  $k = l = 4$  for  $n = 50$ , and  $k = l = 10$  for  $n = 100, 400$ . It uses inflation with  $\delta = 0.02$  for  $n = 10, 20, 50$  and  $\delta = 0.01$  for  $n = 100, 400$ . The EnKF uses Cohn-Gaspari covariance tapering with  $c = 5, 10$  for  $n = 10, 20$ , and  $c = 20$  for larger ensemble sizes. The inflation rate  $\delta = 0.04, 0.04, 0.02, 0.02, 0.01$  for  $n = 10, 20, 50, 100, 400$ , respectively. For NLEAF1, the localization parameter is  $k = l = 3$  for  $n = 10, 20$ , and  $k = l = 4$  for  $n = 50, 100, 400$ , respectively; the inflation rate  $\delta = 0.06, 0.04, 0.02, 0.01, 0.01$  for  $n = 10, 20, 50, 100, 400$ , respectively. NLEAF1q only works for  $n = 400$  with  $k = l = 4$  and inflation rate 0.09. All methods use inflation as described in (7).

(ii) Results

Results on the average RMSE are summarized in Fig. 1 for a quick comparison. Table 4 summarizes some further comparisons on ensemble spread and confidence interval coverage. The ensemble spread is measured by the square root of trace of the ensemble covariance matrix, scaled by  $1/\sqrt{40}$  to match the RMSE. The number below is the percentage of  $x_t^*(1)$  being covered by

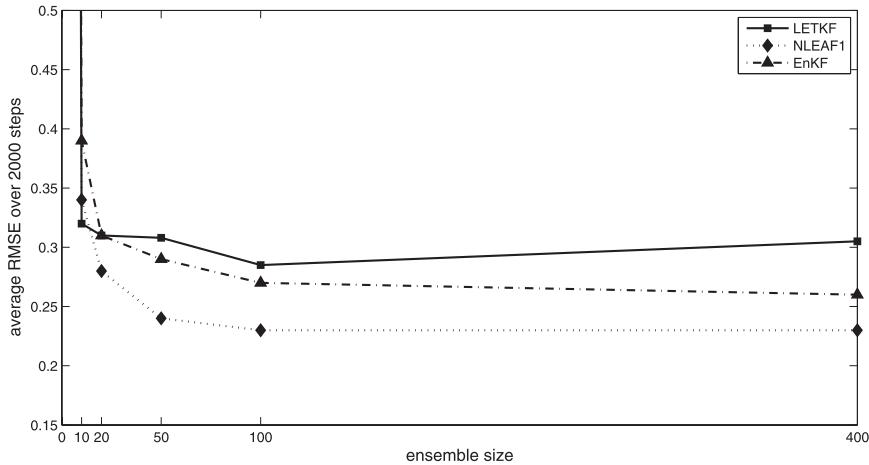


FIG. 1. Ensemble size against average RMSE for the Lorenz-96 system with step size 0.05 and double-exponential observation noise.

the 95% confidence interval. A not applicable (N/A) entry indicates that the filter diverges (the whole ensemble is far away from the true orbit).

From Fig. 1 and Table 4 we see that the NLEAF1 performs better than the LETKF as long as the ensemble size exceeds 20, providing both accurate point estimates and confidence intervals. In this setting the system is pretty linear and NLEAF1 outperforms the LETKF mainly because of the non-Gaussian observation noise. One might notice that the LETKF performs slightly worse for  $n = 400$  than  $n = 100$ . This could be a result of its vulnerability to the presence of outliers (Lawson and Hansen 2004; Lei et al. 2010). That is, increasing the sample size also increases the chance of having outliers, which has a significant impact on the performance of deterministic filters.

### 5. Further discussion

This article demonstrates how simple statistical ideas can help design better filtering algorithms. The NLEAF algorithm inherits some key features of the EnKF by

keeping track of each particle, which makes it suitable for high-dimensional problems with spatial structures such as the Lorenz-96 system. It also reflects the bias-variance trade-off principle: the EnKF is computationally stable but biased for nonlinear non-Gaussian problems, whereas the particle filter is consistent but unstable for moderate-scale problems. The NLEAF is somewhere in the middle: it uses the particle filter only up to the first and second moments, while avoiding reweighting/resampling to maintain stability.

Although the NLEAF works nicely for the Lorenz models, it does not completely overcome the collapse problem of the particle filter. Currently the implementation of NLEAF depends on the sliding-window localization. One might expect difficulties if the state vector is only sparsely observed. In our experiments the largest local window is of size 9, which is still a small number for real-world systems such as a global climate model. For higher dimensionality, importance sampling is likely to collapse even for the first moment. One potential solution to this problem could be to assimilate each observation

TABLE 4. Average RMSE over 2000 time steps for the Lorenz-96 system with complete observation and step size  $\Delta = 0.05$ . The observation errors are sampled from a double-exponential distribution with zero mean and scale parameter  $\theta = 1$ . Also presented is the average ensemble spread (in parentheses) and the percentage of  $\mathbf{x}_t^*$  (1) being ranked between 2.5% and 97.5% in the updated ensemble (below the average RMSE and standard deviation).

Ensemble size	10	20	50	100	400
LETKF	0.32 (0.35) 90.6	0.31 (0.36) 95.0	0.31 (0.36) 96.7	0.28 (0.32) 96.9	0.31 (0.33) 95.8
NLEAF1	0.34 (0.30) 78.7	0.28 (0.30) 92.1	0.24 (0.26) 93.8	0.23 (0.25) 95.5	0.23 (0.25) 95.6
NLEAF1q	N/A	N/A	N/A	N/A	0.39 (0.37) 93.9
EnKF	0.39 (0.38) 83.8	0.31 (0.32) 89.4	0.29 (0.30) 93.5	0.27 (0.27) 95.5	0.26 (0.29) 96.5

coordinate sequentially, with the impact on each state coordinate adjusted according to its distance to the observation (Anderson 2003, 2007).

The NLEAF algorithm originates from a regression perspective of the EnKF. Such a perspective can be used in other ways to design new filtering methods. In some high-dimensional problems, the sparsity structure (or the support) of the state covariance matrix may not depend only on the physical distance. As a result, the sliding-window localization method described in section 4b is no longer valid. Then one may be able to employ sparse regression methods to identify the set of “effective” neighborhood for each state coordinate.

Another direction of future work is the development of a deterministic NLEAF. The NLEAF update introduced here uses perturbed observations that are subject to sampling errors. It is possible to reduce such sampling errors in a deterministic NLEAF update where the posterior moments are estimated by importance sampling or nonlinear regression methods, but the ensemble is updated deterministically to have the corresponding moments.

*Acknowledgments.* The authors thank Jeff Anderson, Thomas Bengtsson, Doug Nychka, and Chris Snyder for helpful comments. Lei and Bickel are supported by Grants NSF DMS-0605236 and NSF DMS-0906808.

#### REFERENCES

- Anderson, J. L., 2001: An ensemble adjustment Kalman filter for data assimilation. *Mon. Wea. Rev.*, **129**, 2884–2903.
- , 2003: A local least squares framework for ensemble filtering. *Mon. Wea. Rev.*, **131**, 634–642.
- , 2007: Exploring the need for localization in ensemble data assimilation using a hierarchical ensemble filter. *Physica D*, **230**, 99–111.
- Bengtsson, T., C. Snyder, and D. Nychka, 2003: Toward a nonlinear ensemble filter for high-dimensional systems. *J. Geophys. Res.*, **108** (D24), 8775, doi:10.1029/2002JD002900.
- Bishop, C. H., B. Etherton, and S. J. Majumdar, 2001: Adaptive sampling with the ensemble transformation Kalman filter. Part I: Theoretical aspects. *Mon. Wea. Rev.*, **129**, 420–436.
- Brusdal, K., J. M. Brankart, G. Halberstadt, G. Evensen, P. Brasseur, P. J. van Leeuwen, E. Dombrowsky, and J. Verron, 2003: A demonstration of ensemble-based assimilation methods with a layered OGCM from the perspective of operational ocean forecasting systems. *J. Mar. Syst.*, **40–41**, 253–289.
- Burgers, G., P. J. van Leeuwen, and G. Evensen, 1998: Analysis scheme in the ensemble Kalman filter. *Mon. Wea. Rev.*, **126**, 1719–1724.
- Chorin, A., and X. Tu, 2009: Implicit sampling for particle filters. *Proc. Natl. Acad. Sci. USA*, **106**, 17 249–17 254.
- Evensen, G., 1994: Sequential data assimilation with a non-linear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *J. Geophys. Res.*, **99** (C5), 10 143–10 162.
- , 2003: The ensemble Kalman filter: Theoretical formulation and practical implementation. *Ocean Dyn.*, **53**, 343–367.
- , 2007: *Data Assimilation: The Ensemble Kalman Filter*. Springer, 279 pp.
- Gaspari, G., and S. E. Cohn, 1999: Construction of correlation functions in two and three dimensions. *Quart. J. Roy. Meteor. Soc.*, **125**, 723–757.
- Gordon, N., D. Salmond, and A. Smith, 1993: Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proc.-F*, **140**, 107–113.
- Hammersley, J. M., and D. C. Handscomb, 1965: *Monte Carlo Methods*. Methuen & Co., 178 pp.
- Houtekamer, P. L., and H. L. Mitchell, 1998: Data assimilation using an ensemble Kalman filter technique. *Mon. Wea. Rev.*, **126**, 796–811.
- , and —, 2001: A sequential ensemble Kalman filter for atmospheric data assimilation. *Mon. Wea. Rev.*, **129**, 123–137.
- Künsch, H. R., 2005: Recursive Monte Carlo filters: Algorithms and theoretical analysis. *Ann. Stat.*, **33**, 1983–2021.
- Lawson, W. G., and J. A. Hansen, 2004: Implications of stochastic and deterministic filters as ensemble-based data assimilation methods in varying regimes of error growth. *Mon. Wea. Rev.*, **132**, 1966–1981.
- Lei, J., P. Bickel, and C. Snyder, 2010: Comparison of ensemble Kalman filters under non-Gaussianity. *Mon. Wea. Rev.*, **138**, 1293–1306.
- Lorenz, E. N., 1996: Predictability: A problem partly solved. *Proc. Seminar on Predictability*, Reading, Berkshire, United Kingdom, European Centre for Medium-Range Weather Forecasts, 1–18.
- , 2005: Designing chaotic models. *J. Atmos. Sci.*, **62**, 1574–1587.
- Mitchell, H. L., and P. L. Houtekamer, 2009: Ensemble Kalman filter configurations and their performance with the logistic map. *Mon. Wea. Rev.*, **137**, 4325–4343.
- Musso, C., N. Oudjane, and F. Le Gland, 2001: Improving regularized particle filters. *Sequential Monte Carlo Methods in Practice*, A. Doucet, N. de Freitas, and N. Gordon, Eds., Springer-Verlag, 247–271.
- Nakano, S., G. Ueno, and T. Higuchi, 2007: Merging particle filter for sequential data assimilation. *Nonlinear Processes Geophys.*, **14**, 395–408.
- Ott, E., and Coauthors, 2004: A local ensemble Kalman filter for atmospheric data assimilation. *Tellus*, **56A**, 415–428.
- Pham, D. T., 2001: Stochastic methods for sequential data assimilation in strongly nonlinear systems. *Mon. Wea. Rev.*, **129**, 1194–1207.
- Sacher, W., and P. Bartello, 2008: Sampling errors in ensemble Kalman filtering. Part I: Theory. *Mon. Wea. Rev.*, **136**, 3035–3049.
- Sakov, P., and P. R. Oke, 2008a: A deterministic formulation of the ensemble Kalman filter: An alternative to ensemble square root filters. *Tellus*, **60A**, 361–371.
- , and —, 2008b: Implications of the form of the ensemble transformation in the ensemble square root filters. *Mon. Wea. Rev.*, **136**, 1042–1053.
- Snyder, C., T. Bengtsson, P. Bickel, and J. Anderson, 2008: Obstacles to high-dimensional particle filtering. *Mon. Wea. Rev.*, **136**, 4629–4640.
- van Leeuwen, P. J., 2003: A variance-minimizing filter for large-scale applications. *Mon. Wea. Rev.*, **131**, 2071–2084.
- , 2009: Particle filtering in geophysical systems. *Mon. Wea. Rev.*, **137**, 4089–4114.
- , 2010: Nonlinear data assimilation in geosciences: An extremely efficient particle filter. *Quart. J. Roy. Meteor. Soc.*, **136**, 1991–1999.
- Whitaker, J. S., and T. M. Hamill, 2002: Ensemble data assimilation without perturbed observations. *Mon. Wea. Rev.*, **130**, 1913–1924.
- Xiong, X., I. M. Navon, and B. Uzunoglu, 2006: A note on the particle filter with posterior Gaussian resampling. *Tellus*, **58A**, 456–460.