

A non-Gaussian Ensemble Filter for Assimilating Infrequent Noisy Observations

By JOHN HARLIM* and BRIAN R. HUNT, *Department of Mathematics and Institute for Physical Science and Technology, University of Maryland, College Park, MD 20742, USA*

(Manuscript received 3 May 2006; in final form 15 December 2006)

ABSTRACT

We present a modified ensemble Kalman filter that allows a non-Gaussian background error distribution. Using a distribution that decays more slowly than a Gaussian allows the filter to make a larger correction to the background state in cases where it deviates significantly from the truth. For high-dimensional systems, this approach can be used locally. We compare this non-Gaussian filter to its Gaussian counterpart (with multiplicative variance inflation) with the three-dimensional Lorenz-63 model, the 40-dimensional Lorenz-96 model, and Molteni's SPEEDY model, a global model with $\sim 10^5$ state variables. When observations are sufficiently infrequent and noisy, the non-Gaussian filter yields a significant improvement in analysis and forecast errors.

1. Introduction

An ensemble Kalman filter (EnKF) estimates the state of a system from a time-series of noisy observations by minimizing a quadratic cost function in the space spanned by an ensemble of forecast model states. Typically, this ensemble space is much lower dimensional than the model state space. Instead of using the Kalman filter equations, which analytically minimize the quadratic cost function, here we numerically minimize a non-quadratic cost function. Our approach is similar to the maximum likelihood ensemble filter (MLEF) of Zupanski (2005), which minimizes a cost function based on a non-Gaussian observation error distribution with a pre-conditioned conjugate gradient method (Fletcher and Zupanski, 2006). In this paper, we show that using a non-quadratic background error distribution can also improve results.

Now we review the Bayesian probabilistic argument (see Lorenc, 1986) for finding the best estimate of the true state \mathbf{x}^t of a system, such as the atmosphere, given noisy observations

$$\mathbf{y}^o = H(\mathbf{x}^t) + \epsilon^o, \quad (1)$$

at the current and past times, where H is the observation operator, and ϵ^o is the observation error. Typically, the true state \mathbf{x}^t and its underlying dynamics are unknown. We assume that the evolution of \mathbf{x}^t is modelled by a chaotic dynamical system governed by a

deterministic differential equation

$$\frac{d\mathbf{x}^t}{dt} = f(\mathbf{x}^t, \mu), \quad (2)$$

where μ is a vector of parameters.

From the probabilistic point of view, we want not just a specific estimate of \mathbf{x}^t at a given time, but a probability distribution $p(\mathbf{x})$ representing the likelihood that a particular model state \mathbf{x} is equal to \mathbf{x}^t . Assuming that prior observations and the model (2) have yielded a background distribution $p(\mathbf{x})$, the goal of data assimilation is to find the analysis probability distribution $p(\mathbf{x}|\mathbf{y}^o)$ given the current observations \mathbf{y}^o in addition to the background information. The analysis state \mathbf{x}^a is typically chosen to be the mode of this distribution, that is, the most likely state. If the distribution of observation errors is known, then applying Bayes rule gives

$$p(\mathbf{x}|\mathbf{y}^o) \propto p(\mathbf{x})p(\mathbf{y}^o|\mathbf{x}). \quad (3)$$

Kalman filters generally assume Gaussian background and observation error distributions: $p(\mathbf{x}) \sim N(\mathbf{x}^b, \mathbf{B})$ and $p(\mathbf{y}^o|\mathbf{x}) \sim N(H(\mathbf{x}), \mathbf{R})$, respectively. Here, \mathbf{x}^b is the background state obtained by feeding the prior analysis state into (2), \mathbf{B} is the background error covariance matrix and \mathbf{R} is the observation error covariance matrix. Thus, taking the logarithm of (3), maximizing (3) is equivalent to minimizing the cost function:

$$\begin{aligned} J(\mathbf{x}) &= J^b(\mathbf{x}) + J^o(\mathbf{x}) \\ &= \frac{1}{2} (\mathbf{x} - \mathbf{x}^b)^T \mathbf{B}^{-1} (\mathbf{x} - \mathbf{x}^b) \\ &\quad + \frac{1}{2} (\mathbf{y}^o - H(\mathbf{x}))^T \mathbf{R}^{-1} (\mathbf{y}^o - H(\mathbf{x})). \end{aligned} \quad (4)$$

*Corresponding authors.
 e-mail: jharlim@cims.nyu.edu
 Present address: Courant Institute of Mathematical Sciences, New York University, 251 Mercer st., New York, NY 10012, USA.
 DOI: 10.1111/j.1600-0870.2007.00225.x

Notice that the background term $J^b(\mathbf{x})$ is quadratic in \mathbf{x} as a consequence of the Gaussian distribution assumption. For linear H , the observation term $J^o(\mathbf{x})$ is quadratic in \mathbf{x} as well. For a non-linear observation operator H , Kalman filters typically make a linear approximation in order to analytically approximate the minimum of the cost function (see Section 2).

A classical approach that is employed by the National Centers for Environmental Prediction (NCEP) uses the same \mathbf{B} for each analysis; this method is known as 3D-VAR (Courtier et al., 1998). However, the background uncertainty can vary considerably from time to time, so it is desirable to allow \mathbf{B} to vary from one analysis to the next. The extended Kalman filter (Ghil et al., 1981) dynamically varies the background covariance matrix using the linear tangent model of (2). However, this background covariance matrix update is not practical for large models. Evensen (1994) introduced an EnKF, which samples the background distribution with an ensemble of background states $\{\mathbf{x}^{b(i)}, i = 1, \dots, k\}$. The ensemble size k is typically much less than the model dimension m . In the last decade, many variations of EnKF have been introduced (Anderson, 2001; Bishop et al., 2001; Houtekamer and Mitchell, 1998, 2001; Keppenne, 2000; Whitaker and Hamill, 2002; Ott et al., 2004).

In this paper, we introduce a non-quadratic (hence non-Gaussian) convex and symmetric background cost term $J^b(\mathbf{x})$ into the framework of a particular type of EnKF, namely the ensemble transform Kalman filter (ETKF, Bishop et al., 2001), which we re-formulate for this purpose. Our approach can replace the local analysis in the LEKF (Ott et al., 2004) when localization is appropriate. The LEKF is designed to be computationally efficient for large systems, and this is not negatively affected by the non-Gaussian modification, which changes only the computations done within the relatively low-dimensional ensemble space. We choose the non-Gaussian background term $J^b(\mathbf{x})$ to grow linearly as $\mathbf{x} \rightarrow \infty$, corresponding to a distribution with density proportional to $\exp(-J^b(\mathbf{x}))$. Thus, the background distribution decay exponentially as $\mathbf{x} \rightarrow \infty$, which is more slowly than a Gaussian. This allows the filter to weight the observations more heavily than the Gaussian filter does when they disagree significantly from the background. We will show that compared to the Gaussian filter, our non-Gaussian filter yields smaller errors when observations are sufficiently infrequent, in case both with and without model error.

More generally, our point is that within the framework we describe, one can experiment with background probability distributions of arbitrary form within the space spanned by the ensemble perturbations. The particular form we use is advantageous in the scenario we study, but we do not presume it to be optimal.

In Section 2, we derive the ETKF from the cost function (4) and review the localization of Ott et al. (2004). In Section 3, we introduce a non-quadratic $J^b(\mathbf{x})$ into the computational framework of Section 2. In Section 4, we present preliminary results obtained for the three-variable Lorenz (1963)

model and, using a local filter, for both the 40-dimensional Lorenz (1996) model and the SPEEDY model (Molteni, 2003). We show results in a perfect model scenario for all three models, where the ‘truth’ is generated from (2) and the filter uses (2) with the same parameter set μ as its ‘model’. For both Lorenz models, we also show experiments with deterministic model error, where the filter uses a different parameter set. Finally, we conclude with a short summary and discussion in Section 5.

2. Variational Formulation of the Ensemble Transform Kalman Filter

EnKFs approximate the true state \mathbf{x}^t by an ensemble whose mean and covariance represent respectively an estimate of \mathbf{x}^t and the uncertainty in the estimate. In the cost function (4), we replace the background state \mathbf{x}^b by the sample mean $\bar{\mathbf{x}}^b$ of the background ensemble, and the background error covariance matrix \mathbf{B} by the sample covariance matrix

$$\mathbf{P}^b = (k - 1)^{-1} \mathbf{X}^b (\mathbf{X}^b)^T, \quad (5)$$

where k is the number of ensemble members and

$$\mathbf{X}^b = [\mathbf{x}^{b(1)} - \bar{\mathbf{x}}^b | \mathbf{x}^{b(2)} - \bar{\mathbf{x}}^b | \dots | \mathbf{x}^{b(k)} - \bar{\mathbf{x}}^b], \quad (6)$$

is an $m \times k$ matrix with columns representing the background ensemble perturbations. Notice that this approximation is problematic for $k < m$ since \mathbf{P}^b is not a full rank matrix, and hence is not invertible. However, $(\mathbf{P}^b)^{-1}$ is well defined on the ‘ensemble subspace’ spanned by the columns of \mathbf{X}^b . Thus, ensemble Kalman filters minimize the cost function on this subspace where it is well defined.

Let us employ a pre-condition (or a coordinate change) by expressing the deviation of a state \mathbf{x} from the background mean state $\bar{\mathbf{x}}^b$ as a linear combination of the background ensemble of perturbations, that is,

$$\mathbf{x} = \bar{\mathbf{x}}^b + \mathbf{X}^b \mathbf{w}, \quad (7)$$

where the weight $\mathbf{w} \in \mathbb{R}^k$ is to be determined, and we approximate the observation vector corresponding to the model state \mathbf{x} by:

$$H(\mathbf{x}) = H(\bar{\mathbf{x}}^b + \mathbf{X}^b \mathbf{w}) \approx H(\bar{\mathbf{x}}^b) + \mathbf{Y}^b \mathbf{w} \in \mathbb{R}^s, \quad (8)$$

where s denotes the number of observations. Here the i th column vector of the $s \times k$ matrix \mathbf{Y}^b is the deviation of $H(\mathbf{x}^{b(i)})$ from its ensemble average; that is,

$$\mathbf{Y}^b = [H(\mathbf{x}^{b(1)}) - \bar{\mathbf{y}}^b | H(\mathbf{x}^{b(2)}) - \bar{\mathbf{y}}^b | \dots | H(\mathbf{x}^{b(k)}) - \bar{\mathbf{y}}^b], \quad (9)$$

with $\bar{\mathbf{y}}^b = \frac{1}{k} \sum_{i=1}^k H(\mathbf{x}^{b(i)})$.¹ Replacing \mathbf{x}^b with $\bar{\mathbf{x}}^b$ and \mathbf{B} with \mathbf{P}^b , and using (5), (7) and (8), reduce the cost function (4)

¹If H is linear, then $\bar{\mathbf{y}}^b = H(\bar{\mathbf{x}}^b)$, but for non-linear H these quantities are different. One could use either $\bar{\mathbf{y}}^b$ or $H(\bar{\mathbf{x}}^b)$ in (8). We always use $\bar{\mathbf{y}}^b$ in forming \mathbf{Y}^b , so that the sum of the columns of \mathbf{Y}^b is zero.

to:

$$J(\mathbf{w}) = \frac{1}{2}(k-1)\mathbf{w}^T\mathbf{w} + \frac{1}{2}(\mathbf{y}^o - H(\bar{\mathbf{x}}^b) - \mathbf{Y}^b\mathbf{w})^T \mathbf{R}^{-1}(\mathbf{y}^o - H(\bar{\mathbf{x}}^b) - \mathbf{Y}^b\mathbf{w}). \quad (10)$$

That is, we reduce the m -dimensional minimization problem to a k -dimensional minimization problem, assuming the ensemble size k is less than the number of model variables m . Notice that in the \mathbf{w} coordinate system, the background error covariance matrix becomes identity and hence we do not have to invert it.

The minimum of (10) is obtained by setting

$$\nabla J(\mathbf{w}) = (k-1)\mathbf{w} - (\mathbf{Y}^b)^T \mathbf{R}^{-1}(\mathbf{y}^o - H(\bar{\mathbf{x}}^b) - \mathbf{Y}^b\mathbf{w}) = \mathbf{0}. \quad (11)$$

The solution of this equation is the analysis weight vector

$$\mathbf{w}^a = \tilde{\mathbf{P}}^a (\mathbf{Y}^b)^T \mathbf{R}^{-1}(\mathbf{y}^o - H(\bar{\mathbf{x}}^b)), \quad (12)$$

where

$$\tilde{\mathbf{P}}^a = ((k-1)\mathbf{I} + (\mathbf{Y}^b)^T \mathbf{R}^{-1} \mathbf{Y}^b)^{-1}. \quad (13)$$

The analysis error covariance matrix $\tilde{\mathbf{P}}^a$ in the ensemble space is the inverse of the Hessian of the cost function (10) (see Fisher and Courtier, 1995; Zupanski, 2005). The analysis state is obtained by substituting (12) into (7):

$$\bar{\mathbf{x}}^a = \bar{\mathbf{x}}^b + \mathbf{X}^b \mathbf{w}^a = \bar{\mathbf{x}}^b + \mathbf{X}^b \tilde{\mathbf{P}}^a (\mathbf{Y}^b)^T \mathbf{R}^{-1}(\mathbf{y}^o - H(\bar{\mathbf{x}}^b)). \quad (14)$$

In the case that H is linear, equations (13) and (14) are equivalent to the standard Kalman filter equations, which minimize J in closed form.

To complete the analysis, we generate an analysis ensemble of model states whose mean is $\bar{\mathbf{x}}^a$ and whose error covariance matrix in the model space is $\mathbf{P}^a = \mathbf{X}^b \tilde{\mathbf{P}}^a (\mathbf{X}^b)^T$. In this paper, we update the ensemble using

$$\mathbf{x}^{a(i)} = \bar{\mathbf{x}}^a + \mathbf{X}^b \mathbf{W}^{a(i)} = \bar{\mathbf{x}}^b + \mathbf{X}^b (\mathbf{w}^a + \mathbf{W}^{a(i)}), \quad (15)$$

where $\mathbf{W}^{a(i)}$ is the i th column of the symmetric square root \mathbf{W}^a of $(k-1)\tilde{\mathbf{P}}^a$. This symmetric square-root together with (13), (14) and (15) is equivalent to the spherical-simplex ensemble

Algorithm 1 ETKF pseudo-code with Kalman filter equations

1. Generate $\mathbf{x}^{b(i)}$ at the current analysis time by feeding $\mathbf{x}^{a(i)}$ from the previous analysis time into model (2).
 2. Form the background ensemble average $\bar{\mathbf{x}}^b$ and matrix of perturbations \mathbf{X}^b given by (6).
 3. Form \mathbf{Y}^b according to (9).
 4. Evaluate $\tilde{\mathbf{P}}^a$ using (13) and \mathbf{w}^a using (12).
 5. Take the symmetric square root of $(k-1)\tilde{\mathbf{P}}^a$ and call its columns $\mathbf{W}^{a(i)}$.
 6. Compute the analysis ensemble members $\mathbf{x}^{a(i)}$ at the current analysis time with (15).
-

Algorithm 2 ETKF pseudo-code with variational formulation

- 4a. Perform the same steps as Algorithm 1 except replace Step 4 with: 4a. Minimize cost function (10) to obtain \mathbf{w}^a . One may use any unconstrained minimization scheme such as the conjugate gradient method.
 - 4b. Evaluate $\tilde{\mathbf{P}}^a$ as the inverse of the Hessian of cost function (10) at \mathbf{w}^a .
-

transform Kalman filter (Wang et al., 2004); in particular, the transform matrix $\mathbf{T}\mathbf{e}^T$ in that paper is equal to our matrix \mathbf{W}^a .

Now we give step-by-step pseudo-codes for this formulation of the ETKF: one uses Kalman filter formulas (see Algorithm 1) and the other uses a variational approach (see Algorithm 2). Regarding step 4b, the Hessian of (10) is the same at every point, but when we generalize to a non-quadratic cost function, it becomes important where we evaluate the Hessian. Of course, in the quadratic case, Algorithm 1 is computationally faster. However, our main goal is to generalize the ETKF for non-Gaussian background error distributions where the Kalman filter equations do not apply. In the next section, we adopt the framework given in Algorithm 2 for a different background cost function term that is still convex and symmetric, with the same Hessian as the quadratic background term at $\mathbf{w} = \mathbf{0}$. Hereafter, by the Gaussian filter we mean the ETKF implemented by Algorithm 1.

For large systems, local analysis has been used for practical purposes and to suppress spurious correlations, caused by small ensemble size, of the model variables between grid points separated by a large distance (see Houtekamer and Mitchell, 1998, 2001; Keppenne, 2000; Ott et al., 2004). Here, we adopt the localization as in Ott et al. (2004). In contrast with the global analysis, the local analysis performs a separate analysis at each model grid point, using observations only from a local region surrounding the grid point. The analysis ensembles computed at each grid point are combined to form a global analysis ensemble. Therefore, the analysis at each location reflects the observations within its local region, which presumably are the observations most correlated to the model state at that location. The local regions should be large enough that neighbouring grid points use heavily overlapping sets of observations; otherwise, imbalances in the resulting global analysis state (see e.g. Cohn et al., 1998) may be problematic. With this approach, each grid point is updated independently. Thus, the analysis can be performed in parallel, which dramatically reduces the cost of its implementation.

3. Non-Gaussian Filter

In practice, even with a perfect model, EnKFs are suboptimal due to model non-linearity and finite ensemble size.

Nonetheless, even for large systems, one can obtain reasonable results with an ensemble of moderate size (less than 100) by spatially localizing the analysis (Houtekamer and Mitchell, 1998, 2001; Keppenne, 2000; Szunyogh et al., 2005). However, regardless of any localization employed, such as Ott et al. (2004), Houtekamer and Mitchell (1998), or Keppenne (2000), the non-linearity of the model together with the small ensemble size still generally cause the analysis ensemble to underestimate its uncertainty (see e.g. Houtekamer and Mitchell, 1998; Anderson and Anderson, 1999; van Leeuwen, 1999; Whitaker and Hamill, 2002). A common approach to overcome this problem is to multiplicatively inflate the background covariance matrix (Anderson and Anderson, 1999). This type of variance inflation also helps, in a very simple way, to overcome the effect of model error, at least to the extent that the model error projects into the space spanned by the ensemble.

In the ETKF, a way of applying variance inflation is to multiply the background covariance matrix $\tilde{\mathbf{P}}^b = (k-1)^{-1}\mathbf{I}$ in the ensemble space by a constant $1+r$ with $r > 0$. This changes the term $(k-1)\mathbf{I}$ of (13) to $(k-1)\mathbf{I}/(1+r)$. Thus in practice, we use Algorithm 1 with the following equation in place of (13):

$$\tilde{\mathbf{P}}^a = \left(\frac{(k-1)}{1+r}\mathbf{I} + (\mathbf{Y}^b)^T \mathbf{R}^{-1} \mathbf{Y}^b \right)^{-1}.$$

In the variational formulation (see Algorithm 2), the variance inflation changes the first term of the cost function $J(\mathbf{w})$ to

$$J^b(\mathbf{w}) = \frac{1}{2} \frac{(k-1)\mathbf{w}^T \mathbf{w}}{1+r}. \quad (16)$$

In a given scenario, we determine the value of r empirically, tuning it to achieve the best results. If the best value of r is small, this suggests that the Gaussian assumption approximates the background statistics reasonably well. However, the best results are obtained with relatively large r when model errors are significant and/or the observations are infrequent. (In the latter case, both model non-linearity and model error have a greater effect from one analysis to the next.) In such cases, the Gaussian assumption probably does not fit the background statistics well. For such situations, we introduce a new background term by replacing (16) with

$$J^b(\mathbf{w}) = \frac{1}{2} \frac{(k-1)\mathbf{w}^T \mathbf{w}}{1 + \alpha \sqrt{\mathbf{w}^T \mathbf{w}}}, \quad (17)$$

where α is a constant to be determined empirically. Notice that when \mathbf{w} is small, (17) behaves like the quadratic function $\frac{1}{2}(k-1)\mathbf{w}^T \mathbf{w}$, and for large \mathbf{w} , (17) grows close to linearly. Function (17) corresponds to a Gaussian-like background error distribution with longer tails (see Fig. 1 for an illustration). This non-Gaussian error distribution approaches a Gaussian error distribution when $\alpha \rightarrow 0$. On the other hand, the tail distribution gets thicker as α increases. Furthermore, this symmetric function remains convex with the same Hessian $(k-1)\mathbf{I}$ at $\mathbf{w} = 0$ for all α . This non-Gaussian filter can be easily implemented by

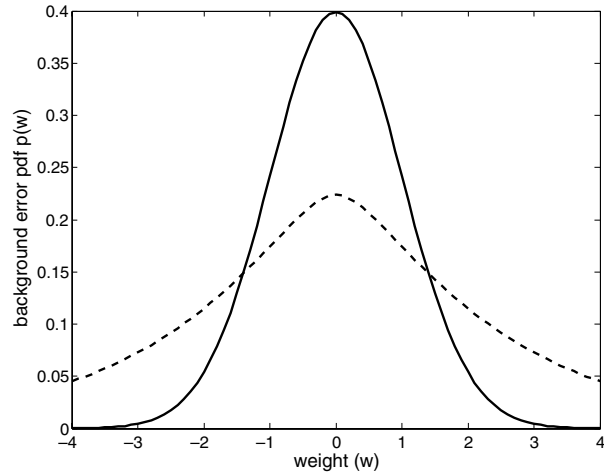


Fig. 1. Density of a one-dimensional Gaussian (solid) error distribution with mean 0 and variance 1 and non-Gaussian (dashed) error distribution with $\alpha = 1$. The standard Gaussian density is proportional to $\exp\{-\frac{1}{2}\mathbf{w}^T \mathbf{w}\}$, while the non-Gaussian pdf is proportional to $\exp\{-\frac{1}{2} \frac{\mathbf{w}^T \mathbf{w}}{1 + \alpha \sqrt{\mathbf{w}^T \mathbf{w}}}\}$.

applying the variational approach (Algorithm 2) to the following cost function instead of (10):

$$J(\mathbf{w}) = \frac{1}{2} \frac{(k-1)\mathbf{w}^T \mathbf{w}}{1 + \alpha \sqrt{\mathbf{w}^T \mathbf{w}}} + \frac{1}{2} (\mathbf{y}^o - H(\bar{\mathbf{x}}^b) - \mathbf{Y}^b \mathbf{w})^T \mathbf{R}^{-1} (\mathbf{y}^o - H(\bar{\mathbf{x}}^b) - \mathbf{Y}^b \mathbf{w}). \quad (18)$$

Notice that when minimizing J , one can easily compute its gradient analytically as in (11).

Based on the discussion above, we see that (18) is very close to (10) when \mathbf{w} is small, but the background term grows more slowly when \mathbf{w} is large. Small \mathbf{w} corresponds to model states close to the background mean $\bar{\mathbf{x}}^b$. Thus, if the background mean agrees well with the observations, both (10) and (18) will be minimized for small \mathbf{w} , and will produce very similar analyses. But when the background mean and observations differ significantly, (18) will be minimized for larger \mathbf{w} , and hence allows a larger analysis increment than (10).

Notice that we still use the linear approximation (8) to the observation operator in (18). When H is non-linear, for greater accuracy one can replace $H(\bar{\mathbf{x}}^b) + \mathbf{Y}^b \mathbf{w}$ with $H(\bar{\mathbf{x}}^b + \mathbf{X}^b \mathbf{w})$ in (18), as in the MLEF (Zupanski, 2005). Using the linear approximation in (18) does have the potential computational advantage that $J(\mathbf{w})$ has a relatively simple algebraic dependence on \mathbf{w} , and we can compute its gradient analytically. In our numerical results below, we consider only cases in which H is linear, so that the only difference between the two filters we compare is the form of the background term.

4. Results

4.1. Numerical simulations with the Lorenz-63 model

In this section, we compare the performance of our proposed non-Gaussian filter with the Gaussian filter described in Section 2 for the three-dimensional Lorenz (1963) model:

$$\begin{aligned} \frac{dx}{dt} &= \sigma(y - x) \\ \frac{dy}{dt} &= \rho x - y - xz \\ \frac{dz}{dt} &= xy - bz, \end{aligned} \tag{19}$$

where the parameter set μ (as in (2)) is the triplet (σ, b, ρ) . We integrate the model using the fourth-order Runge–Kutta method with time step 0.01.

In our numerical simulations, we generate the true state by running the model for 500 non-dimensional time units (i.e. 50 000 steps) with $\mu = (10, 8/3, 28)$, which results in a ‘butterfly’-like attractor Lorenz (1963). With these parameters, the attractor has one positive Lyapunov exponent that corresponds to a doubling time of 0.78 time units. The Kaplan–Yorke dimension is 2.06. Both the leading Lyapunov exponent and dimension are obtained via software ‘Dynamics’ (Nusse and Yorke, 1997). In our simulation, we generate infrequent ‘observations’ every 0.5 time units² by adding Gaussian noise with mean 0 and variance 4 to each coordinate of the true state \mathbf{x}^t . Hence, the observation operator H is linear, equal to the identity and the observation error covariance matrix \mathbf{R} is a diagonal matrix with all diagonal components equal to 4. This choice of error variance in the following experiments implies that a typical observation error is 2; by comparison, the natural variabilities (standard deviations over time) of x, y and z are 7.91, 8.98 and 8.60, respectively.

We consider three cases: perfect model (no model error) by setting the forecast parameter set $\mu_f = \mu$, small model error by setting $\mu_f = (10, 8/3, 30)$, and large model error with $\mu_f = (10, 8/3, 35)$. Table 1 shows the rms difference between analysis and true states for each coordinate, averaged over time and over 10 simulations (where we omit the first 50 analysis steps in each simulation to remove transient behaviour and compute the analysis error for the remaining 950 cycles); we refer to this average rms error as the average analysis error hereafter. Each simulation was based on a different trajectory for the ‘true’ state but we used the same trajectories and associated ‘observations’ for each choice of the filter and model error.

All results use an ensemble of size $k = 10$. In the Gaussian filter experiments, we use $r = 4.5$ for the no model error case, $r = 5.5$ for experiments with small and $r = 10.5$ for large model errors. The non-Gaussian filter simulations with no model error

Table 1. Average analysis error of Gaussian and non-Gaussian filters with the Lorenz-63 model. Here, the average is calculated over 950×10 analysis cycles in RMS sense. The filters are run with ensemble size $k = 10$ and observation error 2. The analysis is performed every 50 steps with time step $\Delta t = 0.01/\text{step}$. In the no model error case, the forecast model parameter set is similar to the true model parameter set, $\mu_f = \mu = (10, 8/3, 28)$. Small and large model errors are introduced by setting $\mu_f = (10, 8/3, 30)$ and $\mu_f = (10, 8/3, 35)$, respectively.

	Gaussian No model error ($r = 4.5$)	non-Gaussian $\alpha = 2$
x	1.38	1.22
y	1.68	1.63
z	1.97	1.63
	Small model error ($r = 5.5$)	$\alpha = 2$
x	1.55	1.33
y	1.84	1.70
z	2.00	1.77
	Large model error ($r = 10.5$)	$\alpha = 10$
x	2.13	1.91
y	2.05	1.84
z	2.26	1.93

and small model error use $\alpha = 2$, and with large model error uses $\alpha = 10$. In each case, these values yielded the smallest average analysis error among the values we tried ($r = 1, 1.5, 2.5, 3.5, 4.5, 5.5, \dots, 10.5$ and $\alpha = 1, 2, 4, 8, 10$). For the large model error, we stop tuning r since we can replace $r \rightarrow \infty$ and the Gaussian filter is equivalent to direct insertion, that is, directly using the observations as analysis, for which the average analysis error would equal the observation error 2. In fact, Table 1 shows that the Gaussian filter with $r = 10.5$ is close to but no better than direct insertion. Table 1 also shows that the non-Gaussian filter performs better than the Gaussian filter by about 10% in all three cases of small, large and no model error experiments.

We measured the variability of the analysis error in time by computing the standard deviation of $|\bar{x}^a - x^t|, |\bar{y}^a - y^t|$ and $|\bar{z}^a - z^t|$. In Table 2, we see that the analysis errors of the non-Gaussian filter are roughly 15% less varied than those of the Gaussian filter in all three cases. For direct insertion, the analysis error variability is about 1.83 in each coordinate, which is significantly larger than both filters in all cases except the Gaussian filter with large model error. Thus the analysis results from the non-Gaussian filter are more consistent in time than those from the Gaussian filter, and both filters performed more consistently than direct insertion.

The filter analysis also shows a further advantage over direct insertion when we consider the forecasts they generate. An important feature of ensemble-based data assimilation is that it

²A full oscillation around one of the ‘butterfly’ wings corresponds to roughly 1 time unit.

Table 2. Variability of the analysis error for the experiments of Table 1. Here, the variability at each coordinate (x , y , and z) is defined to be the temporal standard deviation (of $|\bar{x}^a - x^f|$, $|\bar{y}^a - y^f|$, and $|\bar{z}^a - z^f|$) over 950×10 analysis cycles. For direct insertion this quantity is about 1.8.

	Gaussian No model error ($r = 4.5$)	non-Gaussian $\alpha = 2$
x	0.99	0.81
y	1.08	1.03
z	1.32	1.03
	Small model error ($r = 5.5$)	$\alpha = 2$
x	1.09	0.86
y	1.21	1.11
z	1.24	1.07
	Large model error ($r = 10.5$)	$\alpha = 10$
x	1.36	1.17
y	1.32	1.13
z	1.42	1.15

naturally yields initial conditions for an ensemble forecast, and we find that forecasting from each analysis ensemble member and averaging the forecasts yields better results than making a single forecast from the analysis ensemble mean. Thus for both filters, we measure their forecast error by calculating the rms difference between the true state and the mean of the forecast ensemble, where each ensemble member is generated by feeding each analysis ensemble member to the model (2). In Fig. 2, we show the average forecast errors as functions of time. Here, the average is taken over the same 9500 analysis cycles as in Tables 1 and 2, and all three variables in rms sense. We graph these average errors on a logarithmic scale, so that the distance between two curves represents a relative (percentage) difference between their average forecast errors. We observe that after the first few time steps, the average forecast errors of direct insertion method (dotted) grow faster than those produced by the two filters. In the small and no model error simulations, the average forecast error of direct insertion saturates around 8.51, similar to the climatological error (see top and middle images of Fig. 2). In these cases, notice also that the average forecast errors of the non-Gaussian filter (dashed) remain lower than those of Gaussian filter (solid) at all times. When the model error is large, the non-Gaussian filter analysis error is, on average, only slightly better than that of direct insertion while the Gaussian is not better. However, the average forecast error of direct insertion again grows more quickly than that of the filters. Here, the model error is large enough so that the skill of the ensembles produced from both filters are indistinguishable after time 0.1 unit.

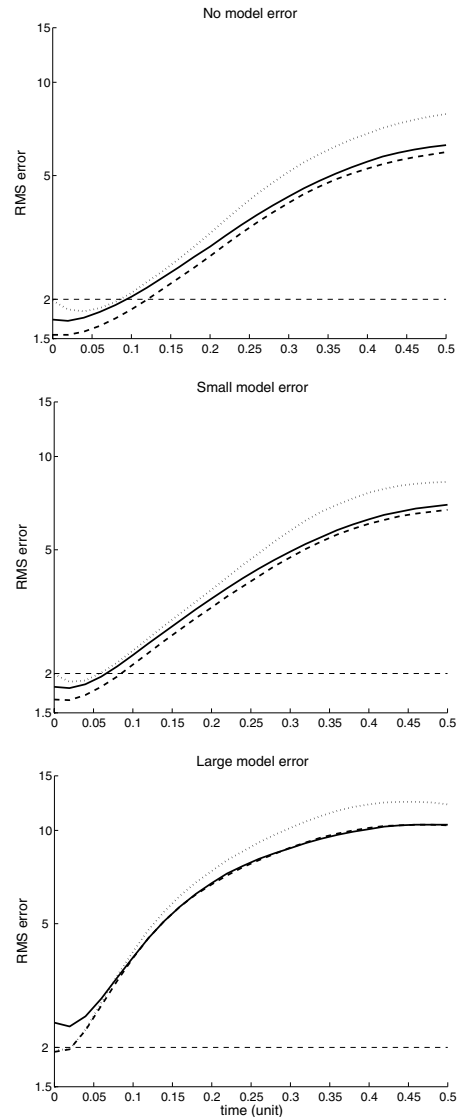


Fig. 2. Average forecast errors as functions of time for the experiments in Table 1: no model error (top), small model error (middle) and large model error (bottom). Each average is taken over 950×10 analysis cycles and 3 model variables in the RMS sense. We use a logarithmic scale on the vertical axis so that the distance between two curves represents the ratio between their errors. The average forecast errors of the initial conditions from direct insertion (dotted) are significantly worse than both the Gaussian (solid) and non-Gaussian (dashed) filters. The observation error is 2 (thin dashed horizontal line).

4.2. Numerical simulations with the Lorenz-96 model

In the previous simulations with the Lorenz-63 model, we saw that our non-Gaussian filter yields better results than the Gaussian filter for a simple temporal chaotic dynamical system, in a case when a large amount of variance inflation is needed for the Gaussian filter. Now, we want to show that this result also holds in a simple spatiotemporal chaotic dynamical system.

For this purpose, we choose the 40-dimensional Lorenz (1996) model and perform a local analysis (as described at the end of Section 2) with the Gaussian and non-Gaussian filters.

The Lorenz-96 model represents an ‘atmospheric variable’ x at m equally spaced points around a circle of constant latitude. The j th component is propagated in time following the differential equation

$$\frac{dx_j}{dt} = (x_{j+1} - x_{j-2})x_{j-1} - x_j + \mu, \quad (20)$$

where $j = 1, \dots, m$ represent the spatial coordinates (‘longitude’). Note that this model is not a simplification of any atmospheric system, however, it is designed to satisfy three basic properties: it has linear dissipation (the $-x_j$ term) that decreases the total energy defined as $V = \frac{1}{2} \sum_{j=1}^m x_j^2$, an external forcing term μ that can increase or decrease the total energy, and a quadratic advection term that conserves the total energy (i.e. it does not contribute to $\frac{d}{dt} V$).

Following Lorenz (1996) and Lorenz and Emanuel (1998), we choose the external forcing to be $\mu = 8$ and the number of spatial elements to be $m = 40$. With these parameters, the attractor has 13 positive Lyapunov exponents, with the leading Lyapunov exponent corresponding to a doubling time of 0.42 time units, and a Kaplan–Yorke dimension of 27.1 (Lorenz, 1996). We use a fourth-order Runge–Kutta scheme for time integration of (20) with time step $\Delta t = 0.05$, and we observe the system state and perform an analysis every six time steps. This is relatively infrequent in the following sense. On the basis of doubling time, Lorenz suggested that 1 time unit of the model is roughly equivalent to 5 d in a global weather model. Thus, performing data assimilation every six time steps of our model integration corresponds roughly to performing it every 1.5 d in a global weather model.

In our numerical simulations, we compute the true state by running the model for 6000 time units (i.e. 120 000 steps). We generate observations every 0.3 time units (six steps) by adding Gaussian noise with mean 0 and variance 1 to each coordinate of the true state \mathbf{x}' . Hence, both the observation operator H and the observation error covariance matrix \mathbf{R} are the identity matrices. For this model the natural variability of each coordinate is 3.61, by comparison with the typical observation error 1. The average analysis error is defined as in the previous section, where the error is averaged temporally in the rms sense by omitting the first 1000 analysis cycles.

We consider two cases: perfect model (no model error) by setting the forecast parameter μ_f to be equal to the true state parameter $\mu = 8$ and with model error by setting $\mu_f = 8.5$. All results use an ensemble of size $k = 10$ and we perform the local analysis at each grid point with localization distance $d = 6$, which has been shown to be optimal for this ensemble size, see Ott et al. (2004); (that is, the local analysis at each grid point uses all observations from $2d + 1 = 13$ grid points centred at the analysis point). In the Gaussian filter ex-

Table 3. Average analysis error and its variability from the experiments with the Lorenz-96 model. Both mean (in RMS sense) and variability are computed over 19000 analysis cycles and 40 variables.

	No model error		Model error	
	Mean	variability	Mean	variability
Gaussian	0.61	0.46	0.68	0.48
non-Gaussian	0.56	0.40	0.65	0.45

periments, we obtained the lowest analysis error with $r = 1.0$ for the no model error case and $r = 1.6$ for the experiment with model error. For the non-Gaussian filter simulations without and with model error, we found the best results with $\alpha = 0.6$ and 0.8, respectively.

In analog to Tables 1 and 2, we show the average analysis error and the analysis error variability in Table 3. Here, the average (in the rms sense) is not only over 19 000 analysis cycles but including the 40 model variables since the model is spatially symmetric, that is, each component possesses an identical dynamics.

Figure 3 shows the average forecast errors as functions of time. Here, the forecast error is averaged over 19 000 analysis cycles and 40 variables in rms sense. Note that the quantities in Table 3 are the respective variables at time 0 in Fig. 3 and 4.

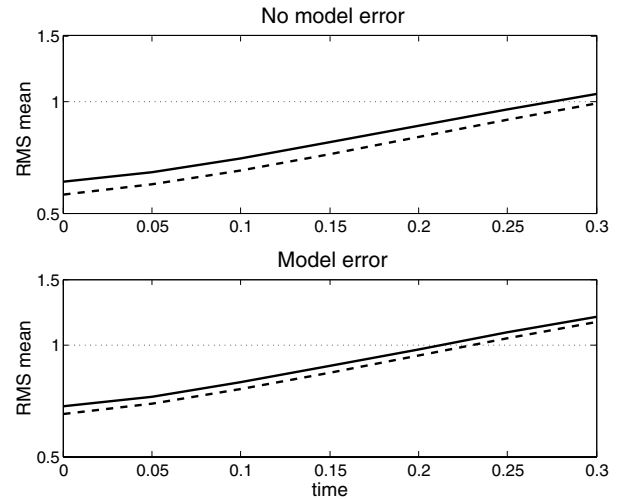


Fig. 3. Average forecast error (again on a logarithmic scale) as a function of time: with no model error (top) and with model error (bottom). Each average is taken over 19 000 analysis cycles and 40 model variables in the RMS sense. The average forecast errors of the Gaussian filter (solid) are larger than those produced by the non-Gaussian filter (dashed) by about 10% in the case of no model error and 5% in the presence of model error. The observation error is 1 (dotted horizontal line). The Gaussian filter uses variance inflation coefficient $r = 1$ and 1.6 without and with model error, respectively. The non-Gaussian filter uses $\alpha = 0.6$ and 0.8 without and with model errors, respectively.

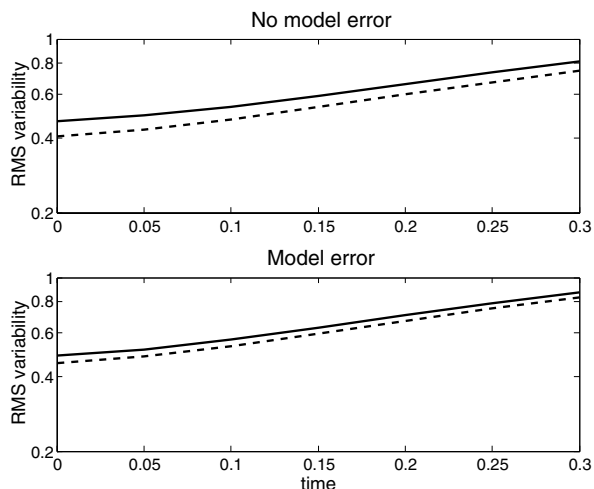


Fig. 4. Forecast error variability (again on a logarithmic scale) as a function of time: with no model error (top) and with model error (bottom). Here, the variability is defined as temporal standard deviation over 19000 analysis cycles and 40 model variables. The forecast error variability of the Gaussian filter (solid) is about 10% larger than those produced by the non-Gaussian filter (dashed).

We observe that the average forecast errors of the non-Gaussian filter (dashed) are about 5% and 10% lower than those produced by the Gaussian filter (solid) with and without model errors, respectively. We also calculate the time variability of the forecast error as a function of time. The forecast error variability is the standard deviation of $\|\bar{\mathbf{x}}^b - \mathbf{x}^t\|$, where $\bar{\mathbf{x}}^b$ denotes the ensemble average of the forecasts produced by propagating each analysis ensemble member as initial condition. We find that the non-Gaussian filter reduces the variation by about 10% (see Fig. 4) both with and without model errors.

4.3. Numerical simulations with the SPEEDY model

In this section, we test both filters on a primitive-equation Global Circulation Model (GCM) in a perfect model scenario. This spectral model (nicknamed SPEEDY, for simplified parametrizations primitive-equation dynamics; see Molteni, 2003, for details) has seven vertical levels (with sigma level 0.950, 0.835, 0.685, 0.510, 0.340, 0.200, 0.080) and a horizontal resolution corresponding to a triangular spectral truncation at total wave number 30 (this yields 96×48 grid points in a standard Gaussian grid). There are five basic prognostic variables: vorticity, divergence, absolute temperature, specific humidity and logarithm of surface pressure. In addition to these variables, the model includes some diagnostic variables (such as saturation specific humidity, relative humidity, dry and moist static energy and saturation moist static energy) whose dynamics follow some simple models of physical processes (such as convection, large-scale condensation, clouds, short-wave and long-wave radiations and diffusion). With these simplified parametrizations, the model is designed to be (at least)

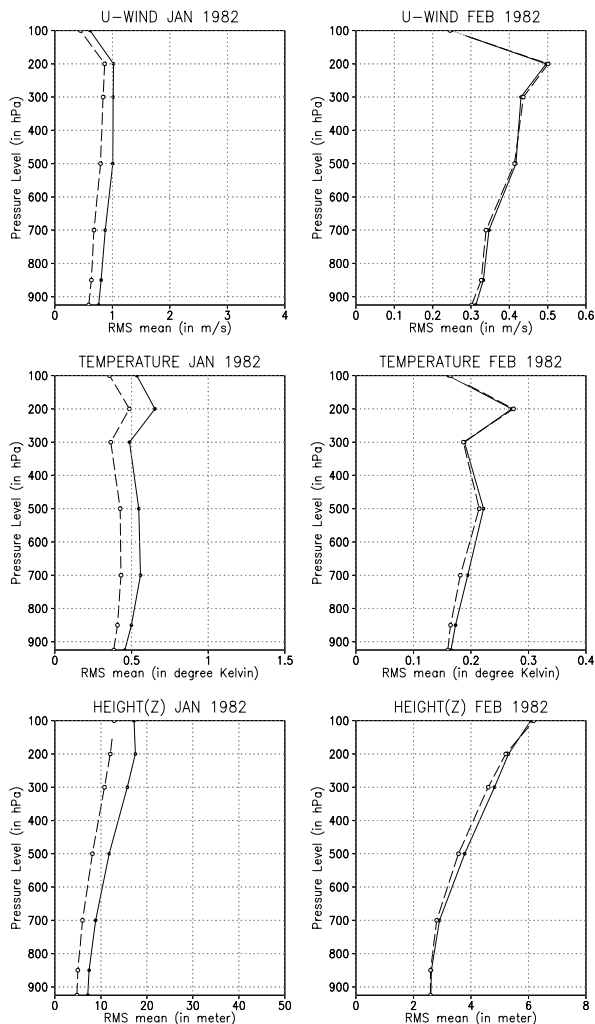


Fig. 5. Average analysis errors as functions of vertical pressure levels (in hPa). The averages (in RMS sense) are taken during January and February and over 96×42 horizontal grid points (excluding latitudes higher than 75°) for variables zonal wind, temperature, and height. Notice that the horizontal scales for January and February differ. In each subfigure, the dashed curve indicates the RMS average from the non-Gaussian scheme with $\alpha = 0.5$ while the solid curve indicates the RMS average from the Gaussian scheme with $r = 20\%$.

an order of magnitude faster (in CPU time) than an operational GCM having similar horizontal resolution.

The model dissipation and external forcing are determined by the following the Northern Hemisphere winter-time climatological fields: sea-surface temperature, surface temperature and moisture in the top soil layer, snow depth, bare-surface albedo, fraction of the sea-ice, land-ice and land-surface covered by vegetation. The prognostic variables are post-processed into zonal and meridional wind components (U-wind and V-wind), absolute temperature (T), specific humidity (Q), geopotential heights (Z) on pressure levels (925, 850, 700, 500, 300, 200, 100 hPa) and surface pressure (Ps).

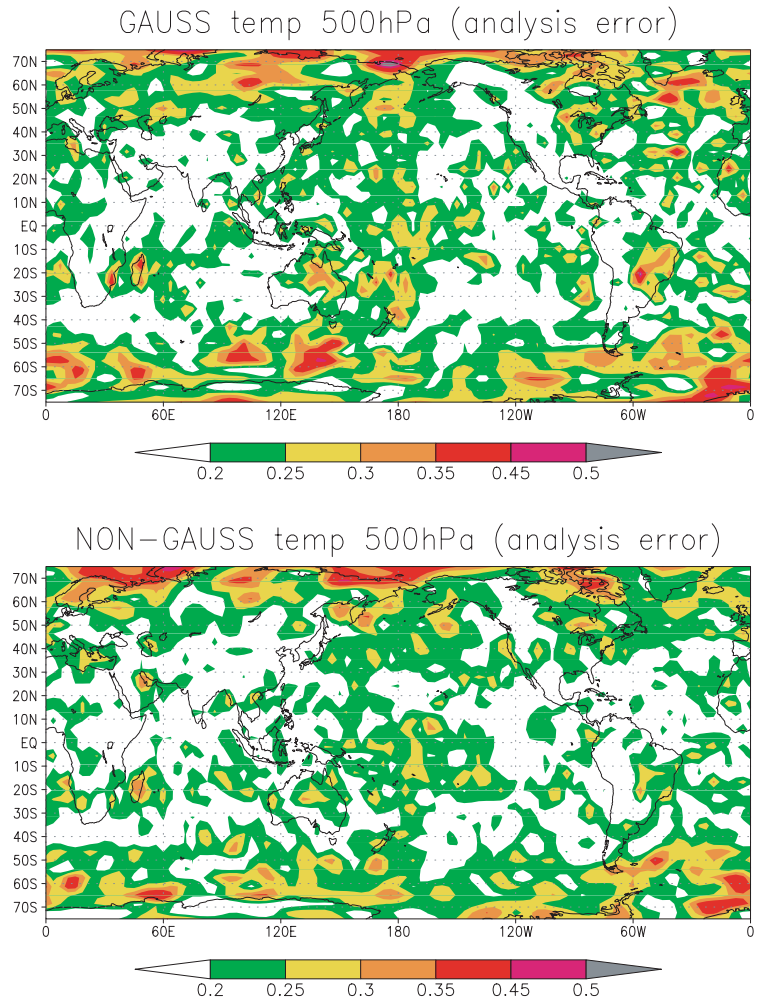


Fig. 6. Average analysis error of temperature (in degrees Kelvin) at 500 hPa during February. The top figure is the Gaussian average analysis error while the bottom one is the non-Gaussian average analysis error.

We compare both schemes with a 24-hr analysis cycle. A true trajectory is created by running the SPEEDY model for 2 months starting from the NCEP reanalysis on 1 January 1982. Then, we generate simulated observations by adding a normally distributed noise with zero mean to the true states every 24 hr. Here the standard deviations of the observation errors for each variable are: 1 m/s for both U and V wind components, 1K for temperature, 0.0001 kg/kg for specific humidity and 100 Pa for surface pressure. By comparison, for pressure level 500 hPa, we compute the natural variabilities of these quantities in the SPEEDY model to be 6.78 m/s for U-wind, 6.84 m/s for V-wind, 2.92K for temperature, 0.0005 kg/kg for specific humidity, while for surface pressure the natural variability is 695.05 Pa. All the results shown in this paper are assimilated with a moderately sparse observation network, that is, 22 % of the grid points (1008 locations) uniformly distributed excluding latitudes higher than 75°N and lower than 75°S at every level.

In each data assimilation experiment, we use a similar initial ensemble of size 20. The initial ensemble consists of states from a long integration of the SPEEDY model at 20 randomly chosen

times. For each local analysis, we use observations from a two-dimensional local region of size 3 × 3 grid points; that is, we use all observations from the same vertical level and up to one grid spacing away in both latitude and longitude. Other local region sizes we tried yield similar or worse results in all the cases we show. In the results shown below, the Gaussian filter (or LETKF) is implemented with variance inflation coefficient $r = 20\%$ after comparing simulations with $r = 10\%, 20\%, \dots, 50\%$. On the other hand, the non-Gaussian filter is shown with $\alpha = 0.5$ after comparing simulations with $\alpha = 0.1, 0.25, 0.5, 1, \text{ and } 1.5$.

Figure 5 shows the average analysis errors of the zonal wind, temperature and height as functions of vertical pressure levels. Here, the averages are calculated over the month of January (31 analysis cycles) and February (28 analysis cycles), and over 96×42 horizontal grid points (weighted according to the grid spacing and excluding latitudes above 75°) in the rms sense. Notice the differences in the rms scales between January and February, the errors in January are much larger than in February due to the initial spin-up time (both filters are initialized with the same random ensemble). We see that the non-Gaussian filter (dashed)

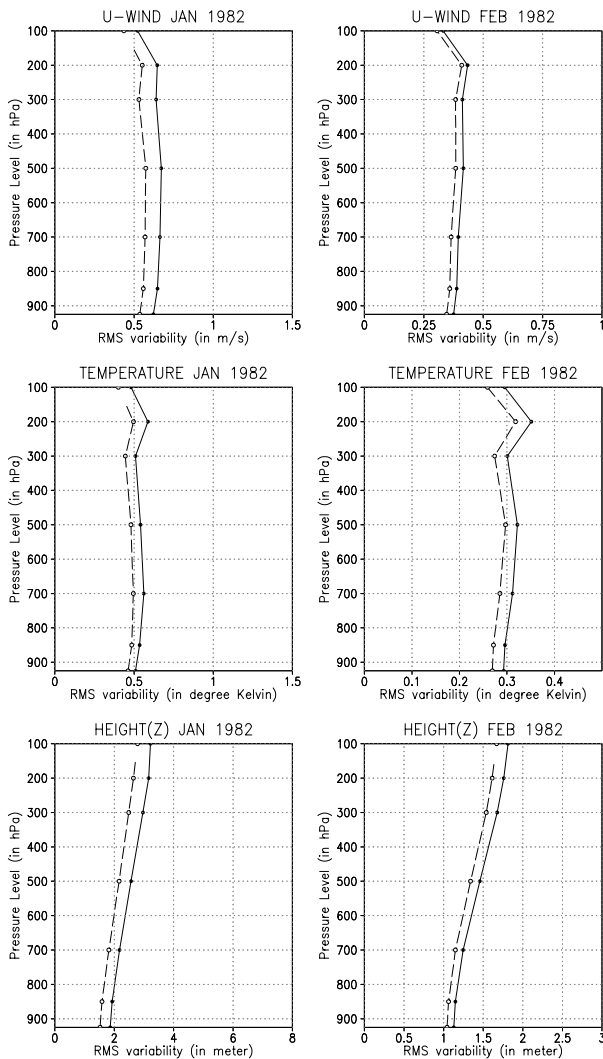


Fig. 7. Analysis error variability as functions of vertical pressure levels (in hPa). The variability is defined as the temporal standard deviation, where the left column is calculated over January while the right column is over February. Notice that the horizontal scales for January and February differ. In each subfigure, the dashed curve indicates the RMS variability from the non-Gaussian scheme with $\alpha = 0.5$ while the solid curve indicates the RMS variability from the Gaussian scheme with $r = 20\%$.

is more effective than the Gaussian filter (solid curves) in January. During the month of February, the rms average indicates that both schemes are comparable. However, if we look at, for example, the temperature field at 500 hPa during the month of February (see Fig. 6), we see that the non-Gaussian filter reduces errors across the southern hemisphere mid-latitude, northern Atlantic ocean, eastern Australia and near Madagascar while it performs worse in relatively small regions such as near the northern hemisphere polar region.

As in Table 2, we also calculate the time variability of the analysis error. In Fig. 7, we plot the variability of the zonal wind,

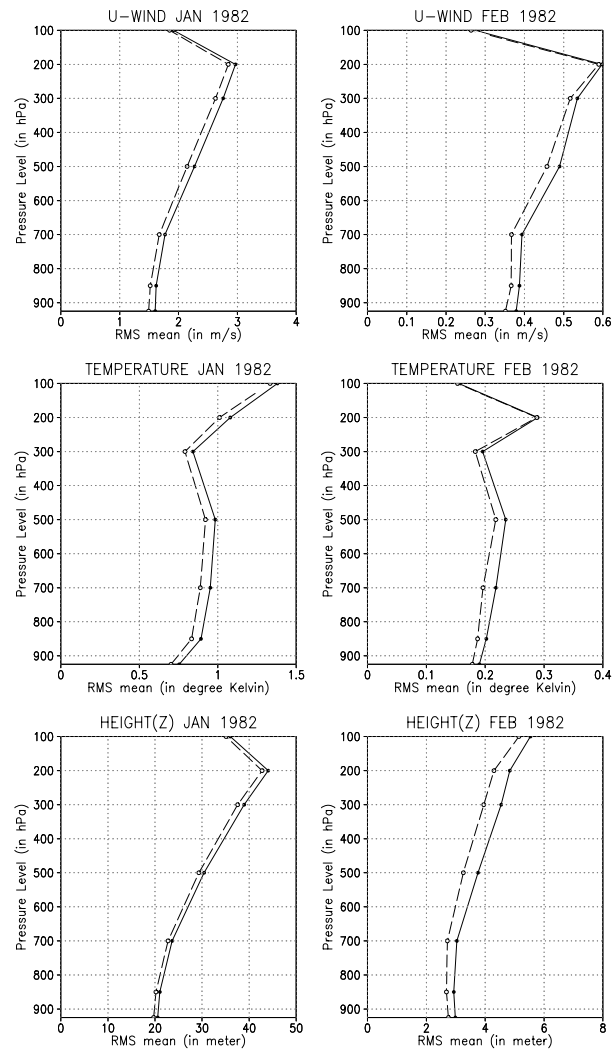


Fig. 8. Average forecast errors as functions of vertical pressure levels (in hPa). The averages (in RMS sense) are taken during January and February and over 96×42 horizontal grid points (excluding latitudes higher than 75°) for variables zonal wind, temperature, and height. The horizontal scales are the same as in Fig. 5. In each subfigure, the dashed curve indicates the RMS average from the non-Gaussian scheme with $\alpha = 0.5$ while the solid curve indicates the RMS average from the Gaussian scheme with $r = 20\%$.

temperature and height as functions of vertical pressure levels. In each period of assimilation, the non-Gaussian analysis (dashed curves) errors are less varied compared to the Gaussian's (solid curves).

In Fig. 8 and 10, the average (24-hr) forecast error and forecast error variability are plotted as functions of vertical pressure levels, respectively. Each average and variability are calculated in the same manner as in Fig. 5 and 7. From these results, we see that the non-Gaussian filter yields a better 24-hr forecast mean, but it only reduces the forecast variability at the later time (February). Figure 9 shows the average 24-hr forecast error of

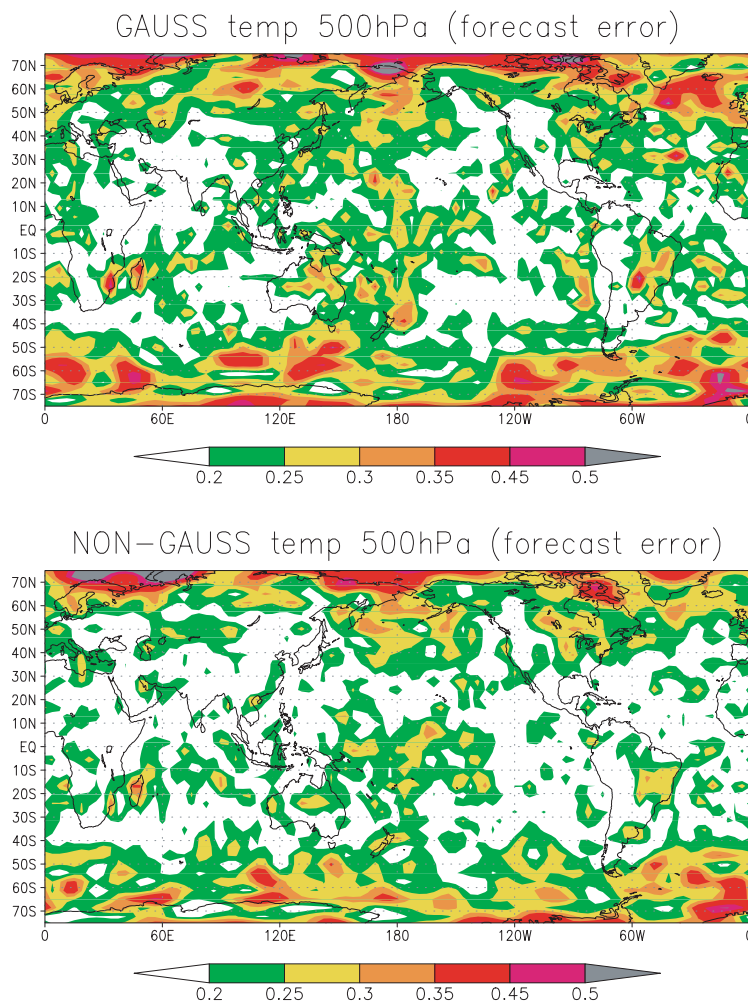


Fig. 9. Average forecast error of temperature (in degrees Kelvin) at 500 hPa during February. The top figure is the Gaussian average forecasts error while the bottom one is the non-Gaussian average forecast error.

the temperature at 500 hPa during February. Here, we see a significant error reduction in those regions mentioned earlier in our discussion of Fig. 6.

5. Summary and Discussion

In this paper, we give a variational formulation of the ETKF and show that a different type of background error distribution (other than Gaussian) can be incorporated within this framework. Specifically, we replace the Gaussian distribution with a non-Gaussian distribution by introducing a non-quadratic background term in the cost function. The symmetric and convex non-quadratic term (17) we use is chosen to closely match the usual quadratic term near its minimum but grow more slowly away from the minimum. This cost term corresponds to a distribution with longer tails than the Gaussian distribution. We compare it to multiplicative variance inflation, which also amplifies the tails of the background distribution but does not change its shape.

In all three simulations described in Section 4, we used observation errors that are a significant fraction (roughly from 1/7

to 1/3) of the natural variability of the observed variables. In the case of the SPEEDY model, the observation errors we used are comparable in size to the errors present in real data. We expect that the difference between the non-Gaussian and Gaussian filters will be larger when the observational errors are larger, and indeed in cases when we tried smaller observation errors, we found less difference between the two methods.

The non-Gaussian and Gaussian filters also produced analyses of more similar quality when we tested them with more frequent observations. When the observations became more frequent, less variance inflation is needed, and as we discussed in Section 3, the two methods become more similar.

We also tested both filters with different ensemble sizes and found that larger ensembles yield somewhat better analyses, but do not significantly affect the comparison between the two schemes (Gaussian and non-Gaussian).

Our results from the Lorenz-63 model simulations show that when the observations are sufficiently infrequent, the non-Gaussian filter yields a significantly better analysis, reducing the analysis error by about 10%. The non-Gaussian filter also

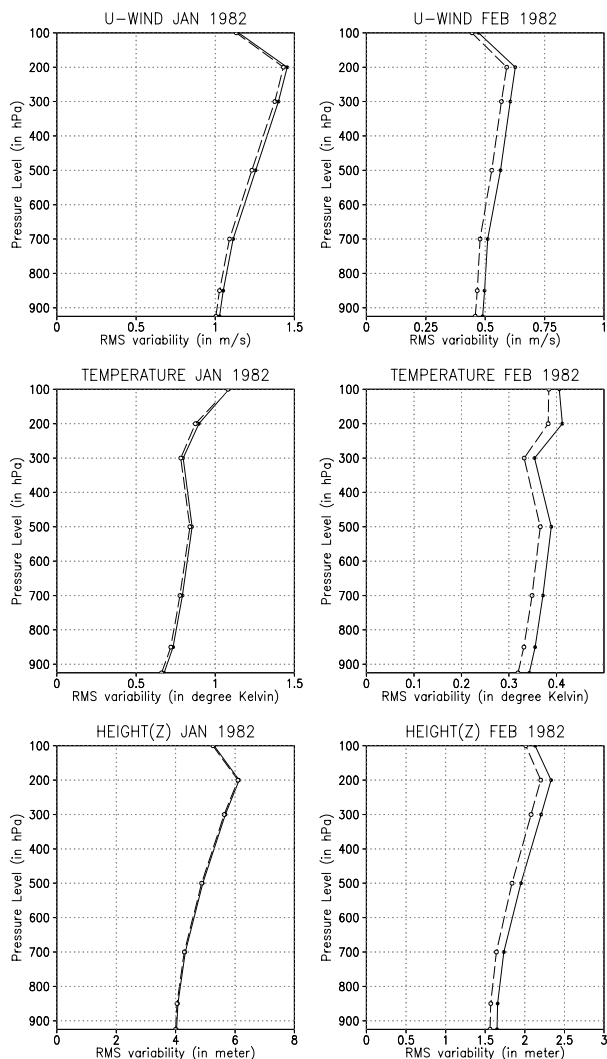


Fig. 10. Forecast error variability as functions of vertical pressure levels (in hPa). The variability is defined as the temporal standard deviation, where the left column is calculated over January while the right column is over February. The horizontal scales are the same as in Fig. 7. In each subfigure, the dashed curve indicates the RMS variability from the non-Gaussian scheme with $\alpha = 0.5$ while the solid curve indicates the RMS variability from the Gaussian scheme with $r = 20\%$.

reduces the analysis error time variations and the forecast errors compared to both the Gaussian filter and direct insertion.

In our simulations with the 40-variable Lorenz-96 model, we performed the analysis locally. We find that with moderately infrequent observations, the non-Gaussian filter yields analysis and forecast errors that are 5–10% lower than those of the Gaussian filter. We also found that the non-Gaussian filter analysis errors are about 10% less variable in time than those of the Gaussian filter.

Our simulation with the SPEEDY model suggests that the non-Gaussian filter has the biggest advantage at the beginning

(in January), when the background ensemble is still far from the true state. During February, both filters produced comparable analyses. The non-Gaussian filter does reduce the time variability of analysis error by 5–10%, and it yields a similar improvement in the February forecast mean and variability. Each data assimilation cycle, excluding model integration, takes about 33% longer to compute for the non-Gaussian filter than for the Gaussian filter.

In the SPEEDY model, we also compared both filters with a denser observation network (results are not shown), that is, when observations are available at each model grid point between 75°N and 75°S (4032 locations) at each level. In this experiment, we found that the analysis and forecast errors are lower for both filters; however, the differences between the two filters are qualitatively similar to the results shown in Section 4 when assimilating with sparse network. However, for the Gaussian filter we need to adjust the variance inflation to $r = 40\%$ for best results, while the non-Gaussian filter still did best with $\alpha = 0.5$. In another experiment (results not shown), we tested both filters with a sparse observation network for a 48-hr analysis cycle and found similar conclusions hold, with the Gaussian filter needing variance inflation of $r = 50\%$, while once again $\alpha = 0.5$ was best for the non-Gaussian filter.

From these experiments, we conclude that except in the case of Lorenz-63 with large model error, the non-Gaussian filter works well with values of α near 1, whereas the optimal values of r for the Gaussian filter varied widely. Thus, we conclude that the parameter α in the non-Gaussian filter is less sensitive to variations of model type and error, observation density and frequency of observations.

Our decision to compare the non-Gaussian scheme with the Gaussian (or ETKF) with multiplicative variance inflation is not because more sophisticated remedies for insufficient ensemble spread are not available, but simply because the multiplicative variance inflation is the most closely related approach to adjusting the background error distribution. Other approaches to overcome the underestimation of the background uncertainty and model errors are discussed, for example, in Mitchell et al. (2002), Tippett et al. (2003), Evensen (2003) and Hamill and Whitaker (2005). It may be useful to combine one of these approaches with a non-Gaussian background term as we describe in this paper.

Finally, as we mentioned in the introduction, one can consider a more general class of non-quadratic terms than (17). For example,

$$\frac{(k-1)\mathbf{w}^T\mathbf{w}}{\gamma + \alpha(\mathbf{w}^T\mathbf{w})^\beta}, \quad (21)$$

include (17), with $\beta = 1/2$ and $\gamma = 1$ and have similar properties for other values of β and γ . In particular, the convexity of this function is retained for $0 \leq \beta \leq 1/2$. The case $\beta = 0$ corresponds to multiplicative variance inflation. We tested several different parameters and found that for $\beta = 1/4$ we did not get a better result than what we showed with $\beta = 1/2$. However, one

may want to further explore these parameters or functions more general than (21) for other models.

6. Acknowledgments

We thank Takemasa Miyoshi for his help with modifying the SPEEDY model code, provided by Franco Molteni, to allow for short-range forecasts; Elana Fertig, Eugenia Kalnay, Eric Kostelich and anonymous reviewers for their constructive criticisms and for reading throughout the manuscript in detail. This research was supported by a James S. McDonnell Foundation 21st Century Research Award, NOAA/THORPEX (grant NA04OAR4310104), and the National Science Foundation (grants DMS 0104087 and ATM0434225).

References

- Anderson, J. L. 2001. An ensemble adjustment Kalman filter for data assimilation. *Mon. Weather Rev.* **129**, 2884–2903.
- Anderson, J. L. and Anderson, S. L. 1999. A Monte Carlo implementation of the nonlinear filtering problem to produce ensemble assimilations and forecasts. *Mon. Weather Rev.* **127**, 2741–2758.
- Bishop, C. H., Etherton, B. J. and Majumdar, S. J. 2001. Adaptive sampling with the ensemble transform Kalman filter. Part I: Theoretical aspects. *Mon. Weather Rev.* **129**, 420–436.
- Cohn, S. E., da Silva, A., Sienkiewicz, M and Lamich, D. 1998. Assessing the effects of data selection with the DAO physical-space statistical analysis system. *Mon. Weather Rev.* **126**, 2913–2926.
- Courtier, P., Andersson, E., Heckley, W., Pailleux, J., Vasiljevic, D., Hamrud, M., Hollingsworth, A., Rabier, F. and Fisher, M. 1998. The ECMWF implementation of three-dimensional variational assimilation (3D-Var). I: Formulation. *Qtr. J. R. Meteorol. Soc.* **124**, 1783–1807.
- Evensen, G. 1994. Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *J. Geophys. Res.* **99**, 10143–10162.
- Evensen, G. 2003. The ensemble Kalman filter: theoretical formulation and practical implementation. *Ocean Dyn.* **53**, 343–367.
- Fisher, M. and Courtier, P. 1995. Estimating the covariance matrix of analysis and forecast error in variational data assimilation. *ECMWF Tech. Memo.* **220**.
- Fletcher, S. J. and Zupanski, M. 2006. A data assimilation method for log-normally distributed observational errors. *Quarterly J. R. Meteorol. Soc.* **132**, 2505–2519.
- Ghil, M., Cohn, S. E., Tavantzis, J., Bube, K. and Isaacson, E. 1981. Applications of estimation theory to numerical weather prediction. In: *Dynamic meteorology: Data assimilation methods*, (eds. L. Bengtsson, M. Ghil and E. Källén), Springer-Verlag, New York, 139–224.
- Hamill, T. M. and Whitaker, J. S. 2005. Accounting for the error due to unresolved scales in ensemble data assimilation: a comparison of different approaches. *Mon. Weather Rev.* **133**, 3132–3147.
- Houtekamer, P. L. and Mitchell, H. L. 1998. Data assimilation using an ensemble Kalman filter technique. *Mon. Weather Rev.* **126**, 796–811.
- Houtekamer, P. L. and Mitchell, H. L. 2001. A sequential ensemble Kalman filter for atmospheric data assimilation. *Mon. Weather Rev.* **129**, 123–137.
- Keppenne, C. 2000. Data assimilation into a primitive-equation model with a parallel ensemble Kalman filter. *Mon. Weather Rev.* **128**, 1971–1981.
- Lorenz, A. C. 1986. Analysis methods for numerical weather prediction. *Quarterly J. R. Meteorol. Soc.* **112**, 1177–1194.
- Lorenz, E. N. 1963. Deterministic nonperiodic flow. *J. Atmos. Sci.* **20**, 130–141.
- Lorenz, E. N. 1996. Predictability: A problem partly solved. In: *Proceeding seminar on predictability*, Vol. 1, ECMWF, Reading, Berkshire, UK, 1–18.
- Lorenz, E. N. and Emanuel, K. A. 1998. Optimal sites for supplementary weather observations: simulation with a small model. *J. Atm. Sci.* **55**, 399–414.
- Mitchell, H. L., Houtekamer, P. L. and Pellerin, G. 2002. Ensemble size, balance, and model-error representation in an ensemble Kalman filter. *Mon. Weather Rev.* **130**, 2791–2808.
- Molteni, F. 2003. Atmospheric simulations using a GCM with simplified physical parametrizations, I: Model climatology and variability in multi-decadal experiments. *Clim. Dyn.* **20**, 175–191.
- Nusse, H. E. and Yorke, J. A. 1997. *Dynamics: Numerical explorations*, Applied Mathematical Sciences **101**. Springer-Verlag, New York, 2nd edition, book includes a computer disk with the program “Dynamics”.
- Ott, E., Hunt, B. R., Szunyogh, I., Zimin, A. V., Kostelich, E. J., Corazza, M., Kalnay, E., Patil, D. J. and Yorke, J. A. 2004. A local ensemble Kalman filter for atmospheric data assimilation. *Tellus* **56A**, 415–428.
- Szunyogh, I., Kostelich, E. J., Gyarmati, G., Patil, D. J., Hunt, B. R., Kalnay, E., Ott, E. and Yorke, J. A. 2005. Assessing a local ensemble Kalman filter: perfect model experiments with the National Centers for Environmental Prediction global model. *Tellus* **57A**, 528–545.
- Tippett, M. K., Anderson, J. L., Bishop, C. H., Hamill, T. M. and Whitaker, J. S. 2003. Ensemble square-roots filters. *Mon. Weather Rev.* **131**, 1485–1490.
- van Leeuwen, P. J. 1999. Comment on “Data assimilation using an ensemble Kalman filter technique”. *Mon. Weather Rev.* **127**, 1374–1377.
- Wang, X., Bishop, C. H. and Julier, S. J. 2004. Which is better, an ensemble of positive-negative pairs or a centered spherical simplex ensemble? *Mon. Weather Rev.* **132**, 1590–1605.
- Whitaker, J. S. and Hamill, T. M. 2002. Ensemble data assimilation without perturbed observations. *Mon. Weather Rev.* **130**, 1913–1924.
- Zupanski, M. 2005. Maximum likelihood ensemble filter: Theoretical aspects. *Mon. Weather Rev.* **133**, 1710–1726.