A Localized Weighted Ensemble Kalman Filter for High-Dimensional Systems

Yan Chen[a], Weimin Zhang[a,b*], Mengbin Zhu[c]

[a]College of Meteorology and Oceanology, National University of Defense Technology, Changsha 410073, China

[b]Laboratory of Software Engineering for Complex Systems, Changsha 410073, China

[c]Beijing Institute of Applied Meteorology, Beijing 100029, China

[*]Correspondence to: Weimin Zhang, College of Meteorology and Oceanology, National University of Defense Technology, Changsha 410073, China. E-mail:wmzhang104@139.com

To avoid filter collapse, a new localized weighted ensemble Kalman filter (LWEnKF) is presented. This filter is a nonlinear non-Gaussian filter that combines some of the advantages of the particle filter (PF) and of the ensemble Kalman filter (EnKF). Additionally, the new method can overcome filter degeneracy in high-dimensional system applications. Based on the weighted ensemble Kalman filter (WEnKF), we extend the scalar weight of each particle to a vector and limit the influence of distant observations through a localization function.

According to the results of experiments using the Lorenz 96 model with 40 variables, LWEnKF with only 10 particles can prevent filter degeneracy. In addition, tests of the new filter are also performed using a two-layer quasi-geostrophic model to demonstrate the feasibility of using the new method in high-dimensional numerical weather prediction models. Comparisons among LWEnKF, the local particle filter (LPF) and the localized perturbed observation EnKF (LEnKF) reveal that the proposed method can combine the advantages of the latter two in certain aspects, even providing better performance in some situations. This characteristic of LWEnKF indicates its potential for data assimilation of different types of observations.

Moreover, the new filter is compared to the block-local ensemble Kalman particle filter. Experiments showed that LWEnKF has an obvious advantage over LEnKPF when the number of particles are small, which indicates its potential for realistic applications limited by computing resources.

Key Words: localization; particle filter; ensemble Kalman filter; proposal density; nonlinear observation operator

1. Introduction

The ensemble Kalman filter (EnKF) and the particle filter (PF) are sequential data assimilation methods based on statistical theory. Compared with PF, EnKF and its derived algorithms have been more widely applied and studied in the field of data assimilation, but note that some of the assumptions adopted by EnKF actually limit the accuracy of analysis. EnKF (Evensen 1994; Burgers et al. 1998) implicitly assumes that the numerical model is linear and that both the forecast error and the observation error are Gaussian-distributed, which is unlikely to be satisfied in most real geophysical systems. In contrast, PF is not constrained by a linear model and Gaussian noise and can be applied to any nonlinear non-Gaussian dynamic system. Additionally, it adopts the Monte Carlo sampling method to approximate the full posterior probability density function (PDF) of model variables and can thus better represent non-Gaussian information.

PF has unique advantages in data assimilation, but there is a drawback. For the generalization of the ensemble idea, PF adds weight to each particle (member of ensemble) to indicate the degree of importance that they have in representing PDFs. When the case is a high-dimensional model with a large number of independent observations, as assimilation progresses forward in time, the weights of a few particles will become increasingly larger, and the weights of most particles will be very small: almost near (or equal to) zero. This occurs even in simple models Farchi and Bocquet (2018). Thus, only a small number of particles (occasionally only one particle) can contribute to characterizing the PDF, while most of them are useless. This is called filter degeneracy, and it directly leads to a low effective sample ratio; thus, the accuracy of the posterior estimation will be greatly reduced, and the forecast clearly cannot be improved, which is one of the main purposes of data assimilation. Resampling is a technique to make weights equal by copying those particles with high weights while abandoning particles with very low weights. However, Snyder et al. (2008) proved that to prevent filter collapse, the number of particles must increase exponentially with the dimension of independent observations, which is the so-called "curse of dimensionality". They also contend that resampling is unable to fully overcome the curse of dimensionality, which is clearly a hindrance to the application of PF in high-dimensional geophysical systems.

To avoid filter degeneracy in practical applications, there are currently three main techniques for improving PF. One technique is to introduce a proposal density that depends on the past model variable and the current observations. Particles are sampled from the proposal density rather than the original transition density. The selection of the proposal density function needs to satisfy a very loose condition: as long as its support contains the support of the original transition density. Theoretically, there are many different choices of proposal densities. Considering a class of proposal density, for each particle, it depends only on the position of the particle at the previous time and the most recent observations. Snyder et al. (2015) argued that filters using sequential importance sampling and any proposal density in the above class cannot avoid the curse of dimensionality, nor can the so-called optimal proposal density that yields minimal degeneracy.

By allowing the proposal density to depend on all particles at the previous time, the equivalent-weights particle filter (EWPF) (Browne and Van Leeuwen 2015) and the implicit equal-weights particle filter (IEWPF) (Zhu et al. 2016) ensure that most or all particles have the same weight, respectively. A comparison of EWPF with the local ensemble transform Kalman filter (LETKF) revealed that the former produced larger root mean squared errors than the latter (Browne 2016). The experimental results of (Zhu et al. 2016) showed that IEWPF has advantages over LETKF in some cases.

Papadakis (2007) used the stochastic EnKF (Burgers et al. 1998; Houtekamer and Mitchell 1998) as the proposal density, proposing a weighted ensemble Kalman filter (WEnKF) (Papadakis 2007; Papadakis et al. 2010). In

†Please ensure that you use the most up to date class file, available from the QJRMS Home Page at
http://onlinelibrary.wiley.com/journal/10.1002/(ISSN)1477-870X

subsequent research, Beyou et al. (2013) used the ensemble transform Kalman filter as the proposal density. Although these two hybrid methods have the potential to combine the advantages of PF and EnKF, Van Leeuwen (2009) noted that they made a mistake in the formulas for calculating weights. However, after the mistake is corrected, these two filters are still not applicable to high-dimensional problems (Van Leeuwen et al. 2015). Localized WEnKF was discussed in Van Leeuwen (2009), but the modifications in this paper were not discussed.

Another technique to relieve filter degeneracy is localization. The idea of calculating the particle weights locally was first introduced by Bengtsson et al. (2003) and Van Leeuwen (2003) and was discussed in detail in (Van Leeuwen 2009). Farchi and Bocquet (2018) divided the localization of PFs into two categories: state-block–domain localization and sequential–observation localization. The idea of state-block–domain localization is similar to that of LETKF. For each grid point, only the observations around it are assimilated, which explicitly achieves localization but inevitably leads to discontinuities between adjacent grid points. Farchi and Bocquet (2018) summarized and presented some approaches to alleviate these discontinuities. The local PF (LPF) proposed by Penny and Miyoshi (2016) uses the state-block–domain localization method and reduces discontinuities via smoothing by weights.

The sequential–observation localization processes the observations in sequence, and only nearby grid points are updated. The algorithms are difficult to parallelize but may mitigate the discontinuities. Depending on the framework of the localized ensemble adjustment Kalman filter (Anderson 2001), Poterjoy (2016) introduced another LPF that uses sequential–observation localization to limit the impacts of observations. LPF expands the scalar weight of each particle to a vector one such that each model variable has a weight. For each observation, LPF only updates the weights of the model variables in the vicinity of the observation and then uses the merging step to modify only the model variables that are close to the observation, while the model variables far from the observation maintain the prior information. The latest research has applied this method to the Weather Research and Forecasting (WRF) model assimilating artificial radar radial velocity and reflectivity observations (Poterjoy et al. 2017). However, Shen et al. (2017) argued that in the calculation formula of local weights, Poterjoy (2016) uses spatially linear interpolation. They proposed a new formulation of the local weights using an exponential tapering of observation influence. However, with the new formulation in Shen et al. (2017), whether the equations of the merging step in Poterjoy (2016) are still appropriate is debatable. In the following, LPF refers to the local particle filter developed by Poterjoy (2016).

As a third way to mitigate filter degeneracy, hybridisation is considered. Taking advantage of combining the characteristics of EnKF and PF, Frei and Künsch (2013) developed the ensemble Kalman particle filter (EnKPF), which was later modified by (Shen and Tang 2015). Different from the idea of WEnKF, EnKPF divides assimilation into two stages with different proportions. The two stages adopt EnKF and PF, respectively, and their proportion is controlled by an adjustable parameter. It constructs a Gaussian mixture model, which can assimilate non-Gaussian errors. By introducing localization, the local EnKPF (LEnKPF) has been applied to a convective-scale numerical weather forecasting model and has assimilated real observations (Robert et al. 2017). Numerical experiments showed that LEnKPF can improve the accuracy of EnKF in strongly nonlinear systems. Combined with anamorphosis and regularisation jitter, the numerical results of the LEnKPF also showed that in a linear system, it is comparable to LETKF; see (Farchi and Bocquet 2018) for details. However, the computational cost of the adaptive choice of the proportion is relatively large.

In this paper, a similar localization approach to LPF is applied to extend WEnKF for high-dimensional systems. The localized EnKF is used to adjust the value of the particles, from which the proposal density is calculated to constitute the total weights. Similar to LPF, the weights are stretched to vector forms, thereby limiting the spatial influence of the observations on the weights. The effectiveness of our new hybrid method was tested by applying it to the Lorenz

$$p(x^n|y^{1:n}) = \frac{p(y^n|x^n)}{p(y^n)} \int \frac{p(x^n|x^{n-1})}{q(x^n|\mathbf{x}^{n-1}, y^n)} q(x^n|\mathbf{x}^{n-1}, y^n) p(x^{n-1}|y^{1:n-1}) dx^{n-1} \quad (1)$$

96 model and the quasi-geostrophic model. The obtained results indicate that localization can effectively solve the filter degeneracy problem and that the new method performs similarly to or even better than LPF or EnKF in both linear and nonlinear cases.

In section 2, we briefly introduce the idea of WENKF, and then we expand the scalar weights of particles to vectors and propose the new local weighted ensemble Kalman filter in section ??. The numerical experiments and results are presented in section 4 before a summary and conclusions are provided in section 5.

2. The weighted ensemble Kalman filter

Similar to the general data assimilation method, we assume that the forecast model is first-order Markov, the observations depend spatial locally on model variables, and the observation errors at different spatial locations or different times are independent.According to Bayesian theory, we multiply and divide the right-hand side by the PDF $q(x^n|\mathbf{x}^{n-1}, y^n)$, which is also called the proposal density. The posterior PDF of the model variable x^n at time-step n given the observations y at time-steps $1, 2, ..., n$ can be written as

Sample x^n from the proposal density $q(x^n|\mathbf{x}^{n-1}, y^n)$ but not the original transition density $p(x^n|x_i^{n-1})$; then,

$$\begin{aligned} p(x^n|y^{1:n}) &= \frac{1}{N} \sum_{i=1}^{N} \frac{p(y^n|x_i^n)}{p(y^n)} \frac{p(x_i^n|x_i^{n-1})}{q(x_i^n|\mathbf{x}^{n-1}, y^n)} \delta(x^n - x_i^n) \\ &= \sum_{i=1}^{N} w_i \delta(x^n - x_i^n) \end{aligned} \quad (2)$$

x_i determines the value of each particle in the model space, and the weight w_i indicates the relative importance of this particle in representing posterior information.

We divide the weight into two parts, with w_i^o and w_i^* denoting the likelihood weight and the proposal weight, respectively.

$$w_i^o = p(y^n|x_i^n) \quad (3)$$

$$w_i^* = \frac{p(x_i^n|x_i^{n-1})}{q(x_i^n|\mathbf{x}^{n-1}, y^n)} \quad (4)$$

The likelihood weight represents the probability density of the observations given a model variable. The proposal weight is a proportion that is related to the use of the proposal density to evolve particles rather than the original transition density; thus, it is related to the choice of the proposal and the forecast model. Because we only need the relative values of weights, on the right-hand side of equation (2), we can ignore the terms with the same value for each particle; then, the total weight is

$$w_i = w_i^o * w_i^* \quad (5)$$

Considering the stochastic EnKF (Houtekamer and Mitchell 2005) as the proposal density, (Papadakis et al. 2010) proposed the weighted ensemble Kalman filter (WEnKF). The formulas for the total weight are given in (Papadakis et al. 2010), but for clarity and completeness of this paper, the derivation of the proposal weight is given in Appendix A. The mistake in (Papadakis et al. 2010) and (Beyou et al. 2013) is that the proposal density is considered to be approximately equal to the original transition density, thus ignoring the proposal weight in the total weight (Van Leeuwen et al. 2015). Consequently, the variance of the total weights is reduced, and the filter degeneracy problem is eased to some extent. However, due to the incorrect calculation of weights, the importance of each particle is misunderstood, and we cannot obtain the correct posterior PDF. The EnKF step used in the WEnKF only makes particles closer to the observations but does not ensure that the weights of particles are evenly distributed. From the current conclusion, using the localized EnKF as the proposal is not valid in high-dimensional systems because it still cannot prevent filter collapse (Morzfeld et al. 2016).

3. The local weighted ensemble Kalman filter

Using the localized EnKF as the proposal is not sufficient to prevent filter degeneracy; thus, we introduce localization to likelihood weight and propose the new local weighted ensemble Kalman filter (LWEnKF). If the observation errors are independent, that is, the covariance matrix R is diagonal, then

$$w_i^o = p(y^n|x_i^n) = \prod_{j=1}^{N_y} p(y_j^n|x_i^n) \tag{7}$$

where y_j is the jth element of the observation vector y. Under this assumption, the observations can be assimilated serially by calculating the likelihood weights one by one. Because the current consideration is the observations at time-step n, we will ignore the superscript n for simplicity.

Following (Poterjoy 2016), we extend the likelihood weights from a scalar to a vector with a length of N_x. Using this vector weight of each particle as a column, we can form an $N_x \times N$ matrix denoted as ω^o. Each element of ω^o is calculated through the formula

$$\omega_{k,i}^o \propto \prod_{j=1}^{N_y} \exp\left\{[p(y_j|x_i) - 1]l_{j,k}^c + 1\right\} \tag{8}$$

where the subscript $k = 1, ..., N_x$ is the model variable indicator. The localization function $l_{j,k}^c$ depends on the spatial location of the observation y_j and the model variable x_k. The cutoff coefficient is denoted by c. We adopt the Gaspri-Cohn (GC) function as the localization function, which is widely used in the localization schemes of EnKFs and PFs. The GC function has compact support and decays to 0 with a radius of $2c$.Note that the total weight is the product of the likelihood weight and the proposal weight; thus, the likelihood weight needs to be normalized before multiplication, which is different from the LPF.

Let $(\hat{\omega}_{k,i}^o)_{y_j} \propto [p(y_j|x_i) - 1]l_{j,k}^c + 1$ be the multiplicative term on the right-hand side of equation (8), while $(\omega_{k,i}^o)_{y_j}$ denotes the weight of $x_{k,i}$ until the observation y_j is assimilated, i.e.,

$$(\omega_{k,i}^o)_{y_j} = (\omega_{k,i}^o)_{y_{j-1}}(\hat{\omega}_{k,i}^o)_{y_j} \tag{9}$$

Similar to (Poterjoy 2016), a parameter $0 < \alpha < 1$ is used to adjust the scalar total weights (4) and the scaler likelihood weights (11) . In fact, this adjustment sets the lower limit of the weights, which can mitigate the occurrence of filter degeneracy. Generally, α is slightly less than 1. There are other better-performing methods to avoid filter degeneracy, such as those described by Farchi and Bocquet (2018) and Poterjoy et al. (2019). Considering that the focus of this paper is on how to localize the WEnKF, we will adopt this simple adjustment.

After the likelihood weight is extended to a vector, the total weight is also a vector that needs to be resampled. A merging strategy was proposed by (Poterjoy 2016). We also use this strategy and make appropriate changes to fit LWEnKF. First, a stochastic universal sampling is performed according to the proposal weights to obtain proposal-resampled particles. This resampling is to consider the influence of proposal density to weights. Then, for each of the observations y_j, the scalar total weights of all the particles are calculated and normalized. According to these total weights, a stochastic universal sampling is performed to obtain total-resampled particles. Last, the corresponding vector total weights are evaluated, and the model variables are updated locally by a linear combination of the proposal-resampled particles and the total-resampled particles.

In addition, similar to (Poterjoy 2016), a nonparametric probability mapping approach called kernel density distribution mapping (KDDM) (McGinnis et al. 2015) is used after all the observations at time-step n are assimilated sequentially. As explained by (Poterjoy 2016) and (Farchi

$$r_{1,k} = \sqrt{\frac{(\sigma_k)_{y_j}^2}{\frac{1}{N-1}\sum_{i=1}^{N}\left\{(x_{k,s_i})_{y_{j-1}} - (\bar{x}_k)_{y_j} + d_k[(x_{k,t_i})_{y_{j-1}} - (\bar{x}_k)_{y_j}]\right\}^2}} \tag{6}$$

and Bocquet 2018), the posterior particles are adjusted by KDDM to have the same quantile as the proposal particles. Experiments will test the effectiveness of this approach in LWEnKF.

In conclusion, when assimilating the observation y, the procedure for LWEnKF is as follows:

1. Use the model forecast equation to evolve all the particles to the observation time step.

2. Perform local perturbed EnKF analysis on each particle to obtain the proposal particles x_i.

3. Calculate the proposal weights w_i^*, $i = 1,\ 2,\ ...,\ N$, see equations (4), (33), and (34). Adjust the proposal weights by the parameter α through

$$w_i^* = (w_i^* - 1)\alpha + 1 \tag{10}$$

And calculate the normalization factor $U = \sum_{i=1}^{N} w_i^*$.

4. Perform stochastic universal sampling according to the proposal weights w_i^* to obtain the proposal-resampled indicator of proposal particles: $t_1, t_2, ..., t_N$.

5. For each scalar observation y_j, update the model variables locally and serially as follows:

   1) For $i = 1,\ 2,\ ...,\ N$, calculate the likelihood weights according to y_j

$$(w_i^o)_{y_j} = [p(y_j|x_i) - 1]\alpha + 1 \tag{11}$$

   2) Calculate the total weights and the normalization factor.

$$(w_i)_{y_j} = (w_i^o)_{y_j} * w_i^* \tag{12}$$

$$(W)_{y_j} = \sum_{i=1}^{N} (w_i)_{y_j} \tag{13}$$

   Based on the scalar total weights, perform a stochastic universal sampling. The total-resampled indicator is denoted as $s_1, s_2, ..., s_N$.

   3) For each model variable k in $(1, 2, ..., N_x)$ and each particle i in $(1, 2, ..., N)$, use equation (9) to update the vector likelihood weights $(\omega_{k,i}^o)_{y_j}$, and derive the vector total weights

$$(\omega_{k,i})_{y_j} = (\omega_{k,i}^o)_{y_j} * w_i^* \tag{14}$$

   4) For each k, calculate the normalization factor $(\Omega_k)_{y_j} = \sum_{i=1}^{N} (\omega_{k,i})_{y_j}$.

   5) For each k and each i, update the model variables by merging the proposal-resampled particles and the total-resampled particles:

$$\begin{aligned}(x_{k,i})_{y_j}^u =& (\bar{x}_k)_{y_j} + r_{1,k}\left[(x_{k,s_i})_{y_{j-1}} - (\bar{x}_k)_{y_j}\right] \\ &+ r_{2,k}\left[(x_{k,t_i})_{y_{j-1}} - (\bar{x}_k)_{y_j}\right]\end{aligned} \tag{15}$$

   where $(\bar{x}_k)_{y_j} = \sum_{i=1}^{N} \frac{(\omega_{k,i})_{y_j}}{(\Omega_k)_{y_j}} (x_{k,i})_{y_{j-1}}$ is the weighted average of the proposal particles. The parameters $r_{1,k}$ and $r_{2,k}$ provide the linear combination of the total-resampled particles and the proposal-resampled particles; see equations (16), (17), (6), (18):

$$d_k = \frac{(\alpha - l_{j,k}^c)U}{l_{j,k}^c (W)_{y_j}} \tag{16}$$

$$(\sigma_k)_{y_j}^2 = \sum_{i=1}^{N} \frac{(\omega_{k,i})_{y_j}}{(\Omega_k)_{y_j}} [(x_{k,i})_{y_{j-1}} - (\bar{x}_k)_{y_j}]^2 \tag{17}$$

$$r_{2,k} = d_k r_{1,k} \tag{18}$$

6. After all the observations at time-step n are assimilated sequentially, the KDDM step is performed for each model variable x_k as follows:

   1) Using Gaussian kernels to approximate the posterior densities by weighted proposal particles or by equal-weight updated particles.

$$p^p(x_k) = \frac{\omega_{k,i}}{\Omega_k} \sum_{i=1}^{N} \mathrm{N}(x_{k,i}, \sigma_k) \tag{19}$$

$$p^u(x_k) = \frac{1}{N} \sum_{i=1}^{N} \mathrm{N}(x_{k,i}^u, \sigma_k^u) \tag{20}$$

2) Using the trapezoid rule to form the cumulative distribution functions (cdfs) for proposal particles (cdf_k^p) and updated particles (cdf_k^u).

3) Using cubic spline interpolation to find the cdf values of each proposal particle $c_{k,i}^p = \mathrm{cdf}_k^p(x_{k,i})$.

4) Using cubic spline interpolation to adjust posterior particles $x_{k,i}^a = (\mathrm{cdf}_k^u)^{-1}(c_{k,i}^p)$.

The derivation of equations (16)-(6) is similar to that in (Poterjoy 2016), with only minor changes (see Appendix B). The motivation is that the mean and variance of the posterior ensemble are approximately equal to the weighted mean and weighted variance of the proposal particles when vector weights are used.

The value of the localization function $l_{j,k}^c$ ranges from 0 to 1. When the spatial position of the model variable is close to the observation, the localization function tends to be 1, and $d_k \to 0$; thus,

$$\lim_{l_{j,k}^c \to 1} r_{1,k} = \sqrt{\frac{(\sigma_k)_{y_j}^2}{\frac{1}{N-1}\sum_{i=1}^{N}\left[(x_{k,s_i})_{y_{j-1}} - (\bar{x}_k)_{y_j}\right]^2}} \approx 1 \tag{21}$$

The approximately equal sign of the formula is established because the variance of the total-resampled particles estimates the weighted variance of the proposal particles with total weights. At the same time, we have $\lim_{l_{j,k}^c \to 1} r_{2,k} = 0$. In other words, for model variables whose position is very close to the observation, the updated particles are almost equal to the total-resampled particles according to equation (15). Likewise, when the distance between the model variables and the observations is very large (exceeds $2c$), we derived $l_{j,k}^c = 0$ and $d_k \to \infty$, which leads to

$$\lim_{l_{j,k}^c \to 0} r_{1,k} = 0 \tag{22}$$

$$\lim_{l_{j,k}^c \to 0} r_{2,k} = \sqrt{\frac{(\sigma_k)_{y_j}^2}{\frac{1}{N-1}\sum_{i=1}^{N}\left[(x_{k,t_i})_{y_{j-1}} - (\bar{x}_k)_{y_j}\right]^2}} = 1 \tag{23}$$

When $l_{j,k}^c \to 0$, the posterior variance and mean are equal to the variance and mean based on the proposal weights, respectively. That is, when assimilating an observation, the model variable far from the observation position is updated by the proposal, and the model variable close to the observation location is updated by WEnKF.

The merging step enables the posterior particles to combine the proposal particles and the resampled particles to avoid filter degeneracy. It is also an artificial approach to adjust the posterior mean and covariance. However, the merging step is not necessary, and we can use resampling instead. To compare the merging step and the adjustment-minimising stochastic universal (SU) sampling, experiments are presented in Appendix C.

Note that although the EnKF has a linear/Gaussian assumption, there is no such hypothesis in the likelihood weights calculation or the sampling and merging step. Therefore, this does not prevent LWEnKF from being a nonlinear non-Gaussian filter.

4. Numerical experiments

In this section, several data assimilation tests are used to test LWEnKF for different observation operators, different ensemble sizes and different models.

A simple but computationally fast 40-dimensional Lorenz 96 (L96) model (Lorenz 2006) is used to compare the analysis results of LWEnKF, the LPF by Poterjoy (2016), the localized perturbed observation EnKF by Anderson (2003) and the block-local EnKPF by Robert and Kunsch (2017). Sequential-observation localization is used in these filters. For convenience of presentation, LEnKF and LEnKPF are used to represent the localized perturbed observation EnKF and the block-local EnKPF, respectively.

First, filters are tested under a mildly nonlinear configuration, which can provide a fair comparison between PFs and LEnKF. Then, nonlinearity and non-Gaussianity are introduced, which makes the data assimilation problem difficult for LEnKF, where PFs excel. We also focus on the improvement of LWEnKF and LPF by KDDM, and the experiments are conducted in a 100-dimensional L96 model.

As a further comparison, LWEnKF, LPF and LEnKF are applied to a two-layer quasi-geostrophic model (Pedlosky 1987) to test the effectiveness of the new method.

The localization function used by the filters is the GC function, and the localization length scales are determined experimentally. As mentioned previously, in the proposal density of LWEnKF, the observation error matrix R is inflated by the inflation coefficient, which is chosen experimentally. The parameter α in LPF and LWEnKF is chosen among $0.80 - 1$ experimentally.

Experiments using the L96 model are programmed in the MATLAB language, in which the inflation of background error covariance in LEnKF and LEnKPF is multiplicative with coefficient testing from 1.00 to 1.20. The two-layer quasi-geostrophic model experiments are performed on the Data Assimilation Research Testbed (DART) and use the adaptive prior covariance inflation (Anderson 2007) in LEnKF with the prior standard deviation set to 0.6. To eliminate the influence of the random initial ensembles on the data assimilation, all the following experiments were repeated ten times for different initial ensembles.

4.1. Lorenz 96 model

The L96 model contains N_x variables whose spatial position are evenly located on a circle. The evolution of each model variable over time is controlled by the following differential equations

$$\frac{\mathrm{d}x_k}{\mathrm{d}t} = (x_{k+1} - x_{k-2})x_{k-1} - x_k + F \tag{24}$$

where $k = 1, 2, ..., N_x$, and $x_{N_x+k} = x_k$, $x_{-k} = x_{N_x-k}$. To ensure chaos in model dynamics, the forcing term of the truth model is typically set to $F = 8.0$. The time step of the L96 model is set to 0.05 (6 hr). The truth model runs 10000 time steps to generate the truth and the observations. The observations are simulated by adding random errors drawn from $\mathrm{N}(0, \mathbf{R})$ to the truth, in which $\mathbf{R} = \mathbf{I}$.

4.1.1. A mildly nonlinear configuration

For the first test, the L96 model with 40 variables is used with no model error. All the model variables are observed directly at every time step, which means that the observation operator is $\mathrm{h} = \mathbf{I}$. This configuration provides an advantage to EnKF but would be challenging for PFs. LWEnKF, LPF, LEnKF and LEnKPF are tested with ensemble sizes of 10, 20, 40, 80, 160, and 320.

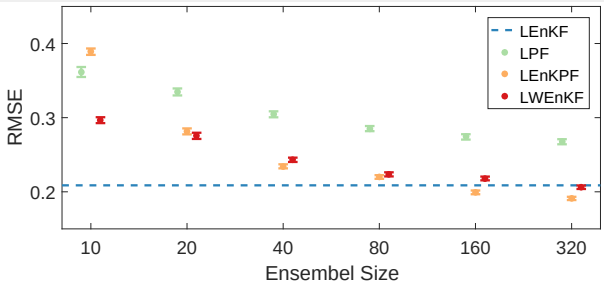

Figure 1. The average posterior RMSE over the data assimilation period using LWEnKF, LPF, LEnKF, and LEnKPF as functions of ensemble size, in which the observation operator is $\mathrm{h}(x) = x$.

The average analysis RMSE is used to evaluate the performance of these filters. The RMSE is averaged over 10000 model time steps. The average RMSEs as a function of ensemble size for each filter are shown in Figure 1. LWEnKF is always significantly superior to LPF. LWEnKF is superior to LEnKPF when the ensemble size is relatively small ($N = 10,\ 20$), while comparable to LEnKF when the ensemble size reaches 320. When the ensemble size is bigger than 160, the RMSEs obtained by LEnKPF are lower than those obtained by the other three filters . LPF abtains higher RMSEs than the other three filters except that it yields a lower RMSE than LEnKPF when the ensemble size is 10.

4.1.2. A nonlinear/non-Gaussian configuration

In this test, the observation occurs every 4 time steps (i.e., 24 hr), and only the model variables at odd positions are observed. The forecast model used in data assimilation experiments has a forcing term of $F = 8.4$, which introduces model error caused by inaccurate model parameters. The model error covariance matrix $\mathbf{Q}$ is calculated statistically using climatological samples of model error. $\mathbf{Q}$ is multiplied by a non-tuning constant to ensure computational stability.

The L96 model with 40 variables is used to test the sensitivity of LWEnKF, LPF, LEnKF and LEnKPF to the ensemble size and the observation type. All filters were tested with ensemble sizes of 10, 20, 40, 80, 160, and 320. Three sets of experiments were conducted, in which the only difference is the specification of the observation operator h. The first set of experiments used a linear function $\mathrm{h}(x) = x$, while

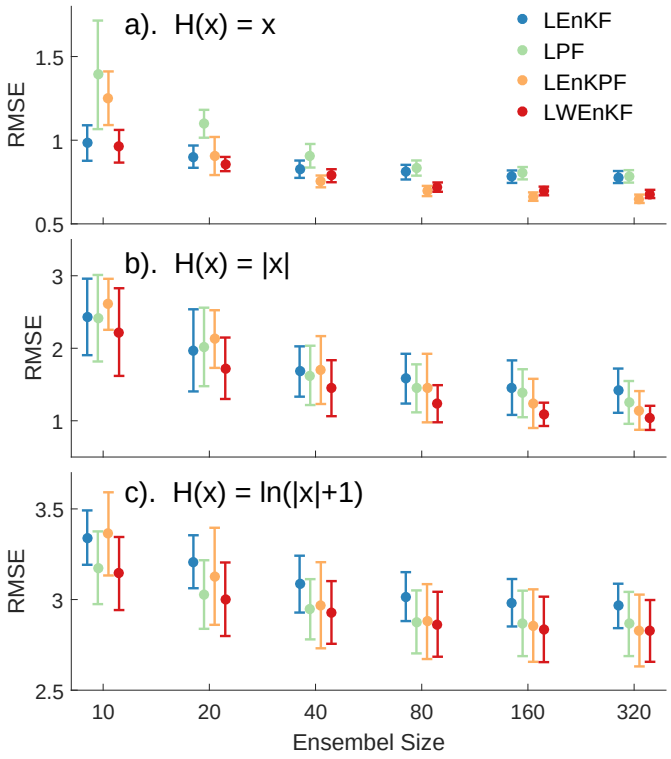

Figure 2. The average posterior RMSE over the data assimilation period using LWEnKF, LPF, LEnKF, and LEnKPF as functions of ensemble size, in which the observation operators are a). $\mathrm{h}(x) = x$, b). $\mathrm{h}(x) = |x|$ and c). $\mathrm{h}(x) = \ln(|x| + 1)$.

the last two sets used nonlinear functions $\mathrm{h}(x) = |x|$ and $\mathrm{h}(x) = \ln(|x| + 1)$, in which the latter has more nonlinearity than the former.

In this test, nonlinearity and non-Gaussianity is introduced in several ways: using rare and sparse observations, using nonlinear observation operators, and using non-Gaussian model errors. These set-ups makes the data assimilation problems easier for PFs than for EnKF.

We used the average analysis RMSE to evaluate the performances of the four filters. The RMSE is averaged over 2500 analysis cycles, i.e., 10000 model time steps. Figure 2 illustrates the average RMSE for three different observation types as a function of ensemble size for each method. In general, we can obtain effective results from LWEnKF for ensemble sizes as small as 10. That is, the filter does not collapse over the entire assimilation period, and the RMSE of the posterior ensemble is smaller than that of the prior ensemble (not shown). Because WEnKF cannot yield these results, the experiments verify that the localization procedure can use small ensembles to prevent filter degeneracy.

When the observation operator is linear in Figure 2 a), LWEnKF can match the performance of LEnKF, while LPF exhibits relatively poor behaviour for the ensemble size tested here. This result occurs because of the Gaussian/linear assumption that LEnKF makes. With more than 20 particles, LEnKPF outperforms the other three filters.

LWEnKF provides an advantage over LPF, LEnKF and LEnKPF when the observation operator is the absolute value of the system variables in Figure 2 b). Nonlinearity of the absolute function fails the Gaussian/linear assumption of LEnKF, but it is not sufficient to fulfil the advantages of PFs; thus, LPF performs worse than LEnKF until the ensemble size reaches 40. When the logarithmic function is taken as the observation operator, as shown in Figure 2 c), the strong nonlinearity allows LPF to outperform LEnKF using as few as 10 particles, while LWEnKF provides results similar to LPF, which confirms its ability to assimilate nonlinear data. With both nonlinear observation operators, the RMSEs obtained by LEnKPF are higher than those of LWEnKF.

For both linear and nonlinear observation operators cases, LWEnKF has an obvious advantage over the other filters when the number of particles are small ($N = 10, 20$) . For more than 40 particles, LEnKPF is slightly better than LWEnKF for the case of the linear observation operator, and for the case of the nonlinear observation operators, it is worse than LWEnKF. LEnKF and LPF have shortcomings in the case of the nonlinear and linear observation operators, respectively, and LWEnKF can be no worse than these two methods under all three observation operator cases. In this sense, we believe that the proposed hybrid method can effectively promote the advantages of LEnKF and LPF, avoiding the disadvantages of the latter two filters to some extent.

4.1.3. Benefits of KDDM

To test the impact of the probability mapping method on LWEnKF, we increase the dimension of the L96 model to a relatively high 100 dimensions. We chose KDDM here as in (Poterjoy 2016). The experiments were conducted with 80 particles. Similar to the set-ups in Section 4.1.2, the model

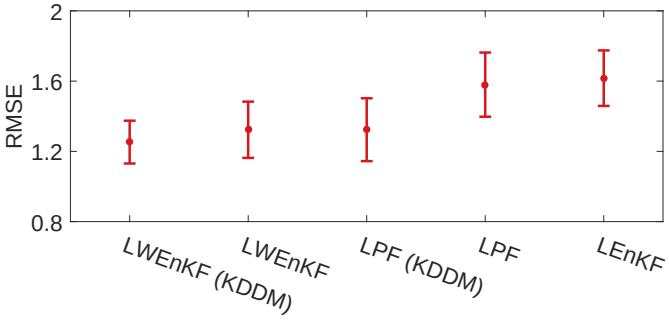

Figure 3. The average posterior RMSE over the data assimilation period using different methods.

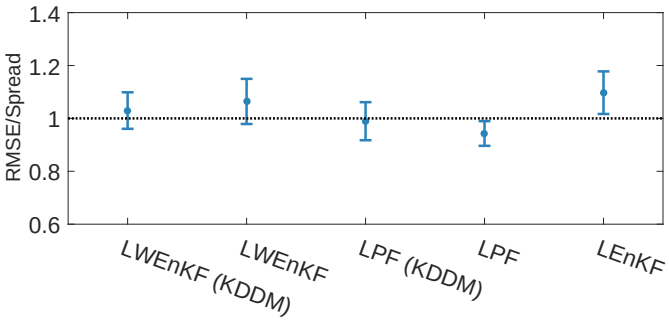

Figure 4. The ratio of the average posterior RMSE to the average ensemble spread for different methods.

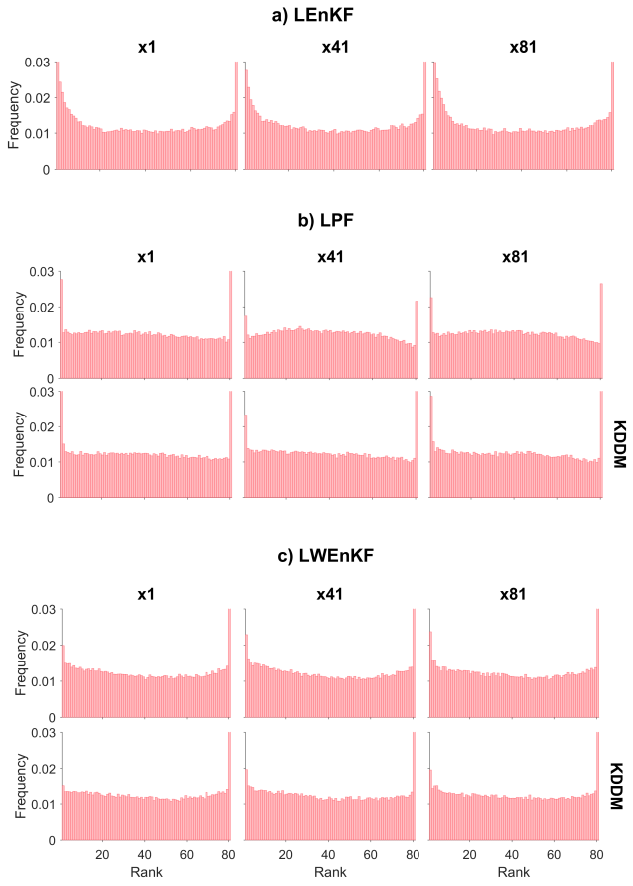

Figure 5. The rank histograms calculated from posterior particles generated by a). LEnKF, b). LPF without (first row) or with KDDM (second row) and c). LWEnKF without (first row) or with KDDM (second row). The columns represent the model variables 1, 21, 41, 61, and 81.

variables at odd positions are observed every 4 time steps, and the observation operator is $h(x) = |x|$. The model error caused by the forcing term $F = 8.4$ is also considered.

Figure 3 shows that when KDDM was applied, both LWEnKF and LPF resulted in a lower RMSE than those without KDDM. The LWEnKF using no KDDM is even more outstanding than the LPF with KDDM. Comparing the ratio of the average RMSE to the average spread illustrated in Figure 4, the spread of LEnKF is too low. When no KDDM step is used, LPF and LWEnKF perform with too much and too little spread, respectively. Applying KDDM causes good adjustment of the relationship between the RMSE and the spread, which can be clearly seen in Figure 4.

As a further comparison, Figure 5 presents rank histograms calculated for every twentieth model variable from LEnKF, LPF and LWEnKF. The abscissa of these figures is discrete bins formed by posterior particles sorted in ascending order, and the ordinate counts the number of times when the true model variable falls within different bins (Hamill 2000). A uniform distribution of count across the bins can reflect that the spread of the analysis ensemble matches the RMSE, which is a good quality of ensembles (Zhu et al. 2016). As shown here, the U-shaped histograms of LEnKF indicate strong under-dispersion. The histograms of the LWEnKF without KDDM are slightly sunken, which can be slightly alleviated by KDDM. The humped histograms of LPF show over-dispersion, which can be modified into a relatively uniform distribution by KDDM. Some of the histograms have high values at both ends, which means that the truth falls outside the range of the particles. This is also happened to most PFs (Farchi and Bocquet 2018). Note that all the histograms are stacked more or less to the left due to model errors, while the histograms of LWEnKF appear to be the most balanced, which reflects the ability of the new algorithm to handle model errors.

The analysis errors of all the variables for the last 1000 time steps are shown in Figure 6. The errors of the LWEnKF with KDDM are far smaller than those of the other experiments for almost all the variables at almost all time steps, which strongly indicates that LWEnKF can significantly outperform LEnKF and LPF with affordable

particles. In addition, even if KDDM is not adopted, the analysis errors of LWEnKF appear to be smaller than those of LEnKF and LPF with KDDM, which further indicates that the hybrid method is better than the other two methods to some extent.

4.2. The two-layer quasi-geostrophic model

The experimental results of the simple model in the previous section indicate that our proposed new hybrid method can combine the advantages of LEnKF for the linear observation operator and LPF for the nonlinear observation operator. This subsection further applies the new method to a quasi-geostrophic model.

The two-layer quasi-geostrophic model considered in this subsection describes the atmospheric flow for geostrophic wind motions. It is a useful tool for studies of data assimilation in NWP systems because it provides some dynamics for operational weather models, such as baroclinic instability. This model supports different resolution settings, has relatively low computational complexity and has been used in studies of the basic characteristics of various data assimilation methods(Bibov et al. 2015; Mussa and Kauranne 2014; Fisher et al. 2011).

The equation and parameter settings of the two-layer quasi-geostrophic model can be found in (Fisher et al. 2011). The time step of the model is 1 hr. The model uses a rectangular grid of 64×32 with 64 zonal grid points and 32 meridional grid points and two vertical layers. The model is added to the DART platform, on which the experiments in this subsection are conducted.

For the perfect model, we set the upper and lower layer depths to $D_1 = 6000$ and $D_2 = 5000$, respectively. The data assimilation tests use a non-perfect model with $D_1 = 5500$ and $D_2 = 4500$. The model variables consist of stream function ψ (dimensionless) and potential vorticity q (dimensionless). The zonal wind u (m/s) and the meridional wind v (m/s) are calculated from the stream function using the centred finite difference method. For simplicity of calculation and programming, the model error covariance matrix is simplified into a diagonal matrix as

$$\mathbf{Q} = \begin{bmatrix} \sigma_u^2\mathbf{I} & & & \\ & \sigma_v^2\mathbf{I} & & \\ & & \sigma_\psi^2\mathbf{I} & \\ & & & \sigma_q^2\mathbf{I} \end{bmatrix} \tag{25}$$

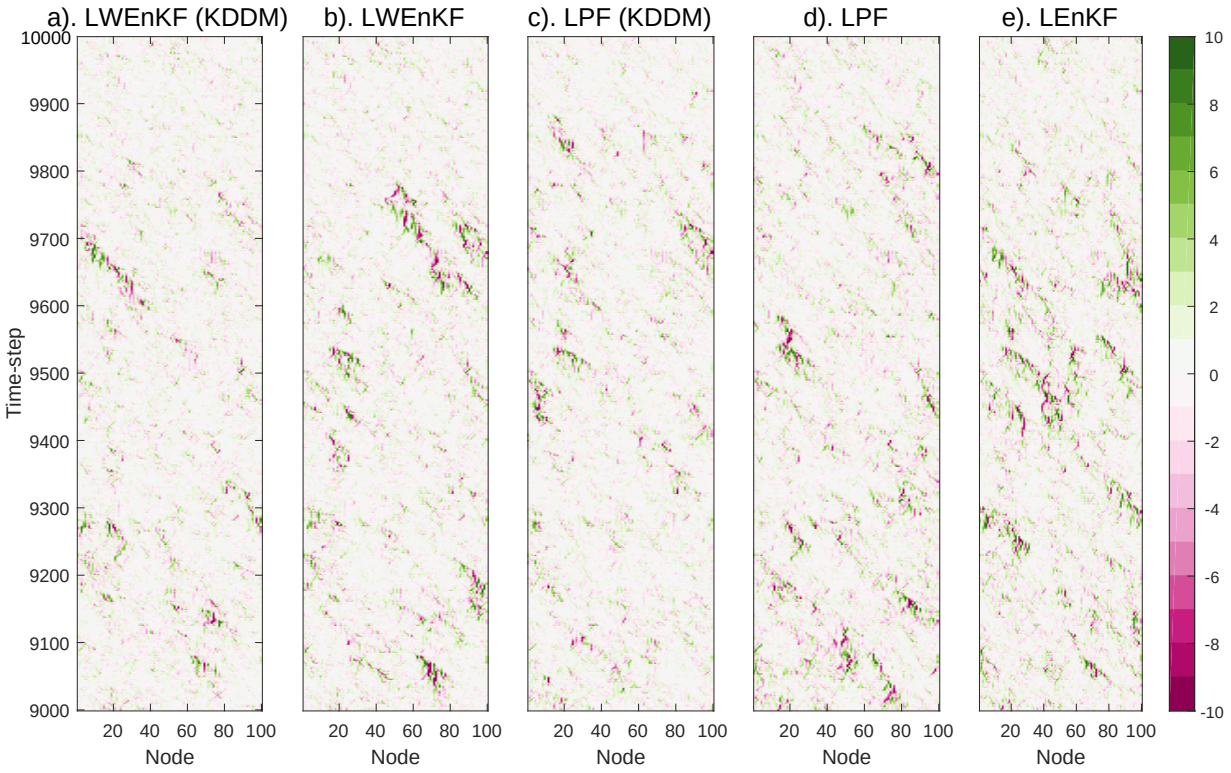

Figure 6. The posterior errors for the last 1000 time steps using a). LEnKF, b). LPF without KDDM, c). LPF with KDDM, d). LWEnKF without KDDM and e). LPF with KDDM.

Table 1. LEnKF, LPF and LWEnKF time-averaged root mean square error (RMSE) and spread of analysis ensemble of each experiment.

RMSE, Spread	$h(x) = x$	$\|x\|$	$\log(\|x - \bar{x}\|)$
	Stream function ψ		
LEnKF	0.4069, 0.3041	0.5772, 0.5375	2.0611, 0.4973
LPF	0.9015, 0.7629	1.0565, 0.8046	1.7954, 1.5133
LWEnKF	0.3989, 0.2942	0.4686, 0.3443	1.6295, 0.8137
	Potential Vorticity q		
LEnKF	2.9881, 1.5766	3.2899, 2.2606	6.3760, 2.3413
LPF	3.7592, 2.3937	4.0428, 2.4628	5.0439, 3.4901
LWEnKF	2.9990, 1.5664	3.2108, 1.7760	5.3939, 2.9349
	zonal Wind u (m/s)		
LEnKF	0.3979, 0.3043	0.5096, 0.4680	1.5996, 0.4719
LPF	0.7366, 0.5924	0.8388, 0.6178	1.3094, 1.0600
LWEnKF	0.3999, 0.2985	0.4620, 0.3413	1.2555, 0.6539
	meridional Wind v (m/s)		
LEnKF	0.3563, 0.2619	0.4658, 0.4138	1.3751, 0.4089
LPF	0.6394, 0.5161	0.7441, 0.5491	1.0932, 0.8771
LWEnKF	0.3530, 0.2572	0.4127, 0.2992	1.0752, 0.5651

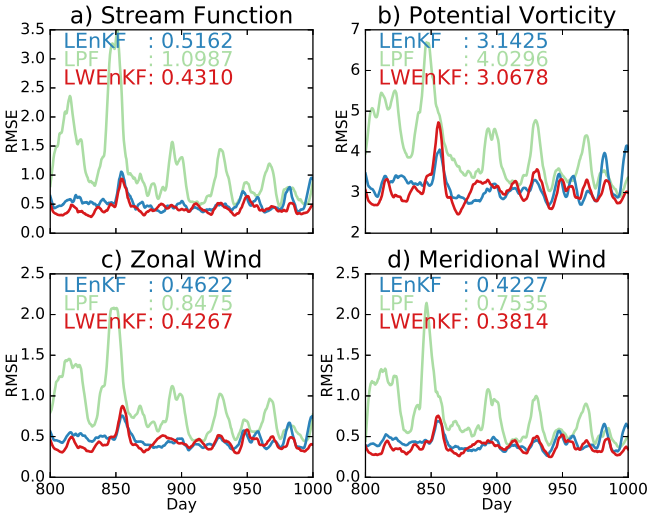

Figure 7. Analysis RMSE of LEnKF (the blue line), LPF (the green line) and LWEnKF (the red line) at every analysis time step for a). steam function, b). potential vorticity, c). zonal wind and d). meridional wind. The values in the upper right corner of each subfigure are the time-averaged RMSEs of the three filters.

Based on a 50-day period model run, the spatially averaged model error standard deviation (σ_u, σ_v, σ_ψ, and σ_q) is statistically estimated using climatological samples. For oceanic or atmospheric data assimilation, the model error covariance can be constructed through control variable transforms.

The model runs a 500 time-step spin up. To generate the initial particles, the model variables are perturbed and run another 200 time steps to adjust the model dynamic balance. Artificial observations are performed every 12 hr at 20 randomly generated positions per layer, and only wind components u and v are observed with Gaussian error $N(0, 0.25)$. Observations are rare and spatially sparse, which simulates real geophysical data assimilation systems. The data assimilation tests were performed for 1000 days, and the first 50 days were discarded as the filter adjustment period. A relatively small 20 particles are used for involving sampling errors.

Inspired by Poterjoy and Anderson (2016), we use three sets of experiments to test LWEnKF, LEnKF and LPF. Each set of experiments uses different observation operators: the linear operator $h(x) = x$ and the nonlinear operators $h(x) = |x|$ and $h(x) = \ln(|x - \bar{x}|)$, where $\bar{x}$ is the climatological mean of the truth model variables.

The RMSE and spread of the analysis particles of the three filters are shown in Table 1. The data in the table are the time averages from 50 days to 1000 days of the assimilation period. LPF and LWEnKF outperform LEnKF in the experiments using logarithmic observation operators, which demonstrates the advantages of the particle filters. LWEnKF yields the lowest RMSE when using the linear and the absolute observation operators. When the nonlinear operator $h(x) = |x|$ is used, the analysis RMSE of LPF is higher than that of LEnKF, which is reasonable due to the small number of particles, which also reflects the shortcomings of LPF. However, LWEnKF alleviates this disadvantage to some extent.

The assimilated observations are the meridional wind and the zonal wind, rather than the model variables potential vorticity and flow function. Comparing the RMSE of the observed model variables with unobserved model variables in Table 1, the ability of the filters to estimate the correlation between multiple variables can be observed. For the potential vorticity and flow function, the performance of LWEnKF is comparable to that of the wind components, which indicates that for multivariate models with complex relationships, LWEnKF can sample in the high-probability region of the posterior PDF.

The following presents some results for experiments using the nonlinear observation operator $h(x) = |x|$. Figure 7 shows the RMSEs of LEnKF, LPF and LWEnKF at each analysis time step, where the RMSE of LWEnKF is the lowest for

most time steps. Moreover, comparing the observed wind components with the unobserved potential vorticity and flow function, the trend of their RMSE time series is similar, which is consistent with the time-averaged RMSE in Table 1.

The truth and comparison of the three filters' analyses in potential vorticity and stream function at 250 days are illustrated in Figure 8. We subtract the truth from the analysis in Figure 8 to obtain Figure 9 such that it can be clearly seen that the analysis field of LWEnKF is the closest to the truth among the three data assimilation methods.

5. Conclusions

This paper presented a localized weighted ensemble Kalman filter, which is a hybrid algorithm that combines the advantages of LEnKF and LPF to some extent and can be applied to high-dimensional systems without filter degeneracy.

LWEnKF first uses LEnKF to assimilate the observations locally and serially and then calculates the proposal weights based on the obtained proposal ensemble and all the observations at the current assimilation time step. Meanwhile, the local likelihood weights are calculated serially. Finally, the sampling and merging are performed according to the product of the likelihood weights and the proposal weights; that is, the particles are updated by the linear combination of the resampling particles and the proposal particles. Optionally, the probability mapping approach is used to adjust the higher moment of the ensemble to obtain the analysis particles.

The new method computes local likelihood weights in vector form. The likelihood weights together with the

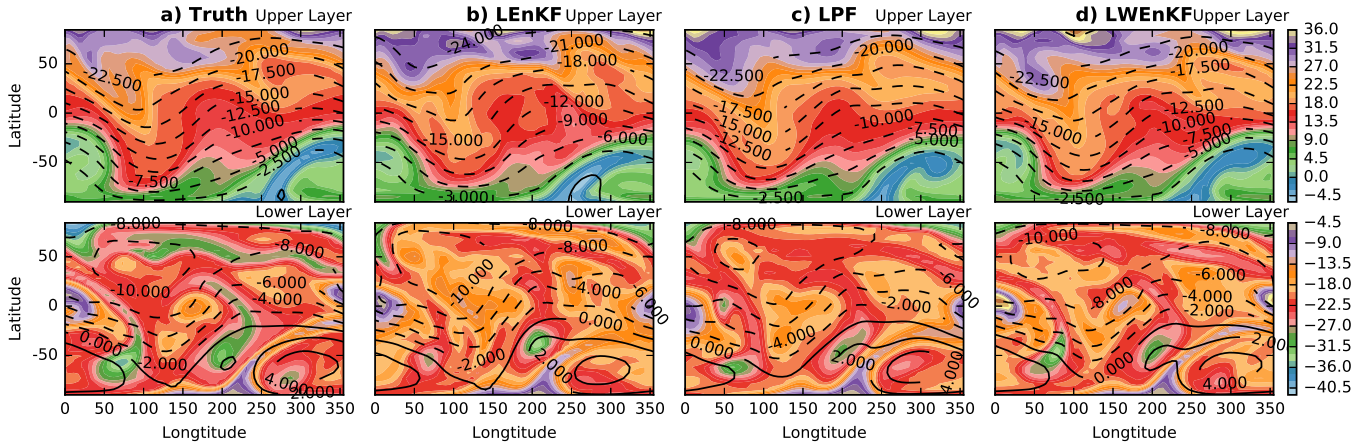

Figure 8. The a). truth and analyses of b). LEnKF, c). LPF and d). LWEnKF in potential vorticity (shade) and stream function (curve, solid for positive values and dotted for negative values) at 250 days.

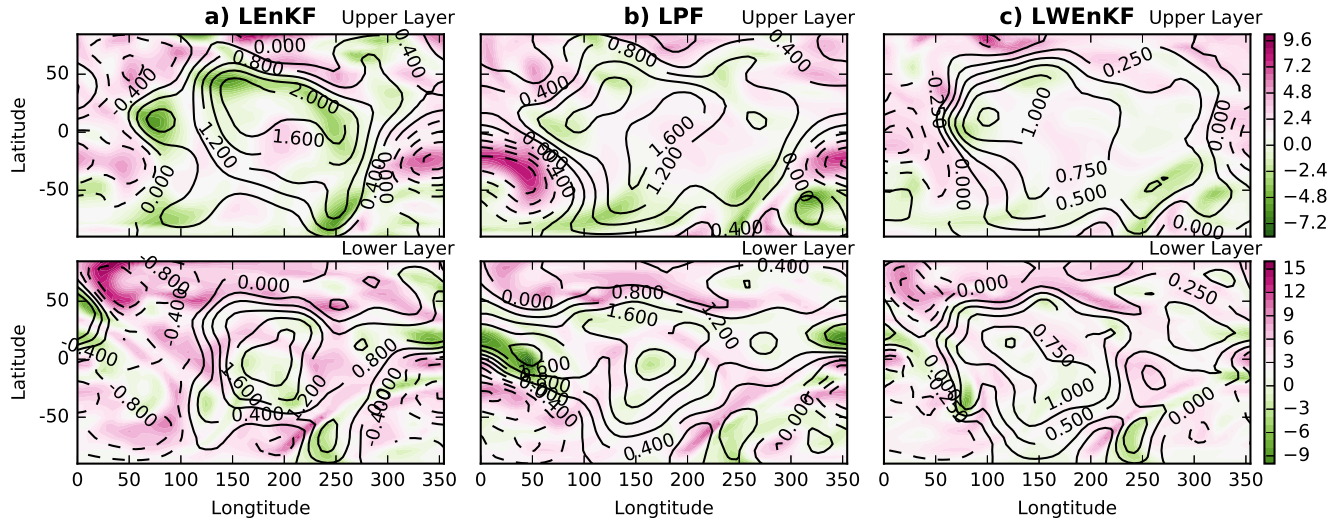

Figure 9. The departures between the truth and analyses of a). LEnKF, b). LPF and c). LWEnKF in potential vorticity (shade) and stream function (curve, solid for positive values and dotted for negative values) at 250 days.

proposal weights constitute the total weights in vector form, which can limit the effect of the observations within the specified spatial domain. In the vicinity of the observations, the updated particles approximate the original WEnKF. Far away from the observations, the particles only retain information from the LEnKF and the proposal weights.

Using the L96 model, we tested the effectiveness of LWEnKF and compared it with LPF, LEnKF and LEnKPF. In the experiments with a mildly nonlinear configuration, LWEnKF is always significantly superior to LPF, while superior to LEnKPF when the ensemble size is relatively small. However, LWEnKF needs a large ensemble size to be comparable to LEnKF with only 10 ensembles. LEnKPF shows significant advantages over LWEnKF when the ensemble size is large.

In the experiments with a nonlinear/non-Gaussian configuration, we set up different observation operators and different scales of particles. It can be found that LEnKF are more efficient than LPF when using the linear observation operator. The more efficient here means that fewer particles are needed to achieve the same assimilation effect (measured by RMSE). When the nonlinearity of the observation operator is strong, LPF can provide better results than LEnKF when acceptable particles are adopted, which is consistent with the conclusions in (Poterjoy 2016). In both the linear and nonlinear observation operator cases, LWEnKF can combine the advantages of LEnKF and LPF and achieve similar or better assimilation efficiency. When the number of particles are small, LWEnKF has an obvious advantage over LEnKPF, LEnKF and LPF. This indicates the potential of LWEnKF for realistic applications, since the atmospheric or oceanic data assimilation systems can only use a few particles due to limited computing resources. Moreover, the KDDM step can improve the relationship between the RMSE and spread of the analysis ensemble, thereby improving the quality of the posterior ensemble.

Furthermore, the new filter is tested in a sufficiently realistic two-layer quasi-geostrophic model. The experimental results show that LWEnKF can effectively avoid the filter degeneracy problem in high-dimensional systems. When the observation operator, the model and the number of particles are the same, compared to LPF and LEnKF, the analysis fields obtained by LWEnKF are the closest to the truth. This result reflects the efficiency of the new approach and its potential for application in numerical weather prediction models.

LWEnKF is added to DART and can easily be combined with various models and can be readily compared with other data assimilation methods. However, the new hybrid method needs to be studied for more realistic geophysical models to test its strengths and weaknesses, which is currently underway.

Appendix A: Formulation of proposal weights

Recall that the assumptions in Section 2 still hold. The equations for the localized stochastic EnKF (Houtekamer and Mitchell 2005) are

$$x_i^{n,f} = \mathrm{M}(x_i^{n-1}) + \beta_i^n \tag{26}$$

$$\beta_i^n \sim \mathrm{N}(0, \mathbf{Q}) \tag{27}$$

$$\epsilon_i \sim \mathrm{N}(0, \mathbf{R}) \tag{28}$$

$$x_i^n = x_i^{n,f} + \mathbf{K}_e[y + \epsilon_i - \mathrm{h}(x_i^{n,f})] \tag{29}$$

where $\mathrm{M}(\cdot)$ is the nonlinear deterministic forecast model and $\mathrm{h}(\cdot)$ is the nonlinear observation operator. The model error β is Gaussian with zero mean and covariance matrix $\mathbf{Q}$. The observation error is denoted as ϵ. The Kalman gain matrix $\mathbf{K}_e$ is calculated from the ensemble and localized.

Considering that the model error is small relative to the model variables, there is an approximation

$$\begin{aligned} \mathrm{h}(x_i^{n,f}) &= \mathrm{h}(\mathrm{M}(x_i^{n-1}) + \beta_i^n) \\ &\approx \mathrm{h}(\mathrm{M}(x_i^{n-1})) + \mathbf{H}\beta_i^n \end{aligned} \tag{30}$$

where H is the tangent linear observation operator corresponding to h. The right-hand side of equation (29) can be divided into two parts: the deterministic part

$$\mu_i^n = \mathrm{M}(x_i^{n-1}) + \mathbf{K}_e[y - \mathrm{h}(\mathrm{M}(x_i^{n-1}))] \tag{31}$$

The remaining part is stochastic. If the model error is not related to the observation error, the covariance matrix of the stochastic part is

$$\mathbf{\Sigma} = (\mathbf{I} - \mathbf{K}_e\mathbf{H})Q(\mathbf{I} - \mathbf{K}_e\mathbf{H})^T + \mathbf{K}_e\mathbf{R}\mathbf{K}_e^T \tag{32}$$

Then, the denominator of the proposal weight (4) is

$$q(x_i^n|\mathbf{x}^{n-1}, y^n) \propto \exp\left[(x_i^n - \mu_i^n)^T\mathbf{\Sigma}^{-1}(x_i^n - \mu_i^n)\right] \tag{33}$$

Finally, it is easy to derive the transition density, which is also the numerator of the proposal weight.

$$p(x_i^n|x_i^{n-1}) \propto \exp[(x_i^n - \mathrm{M}(x_i^{n-1}))^T\mathbf{Q}^{-1}(x_i^n - \mathrm{M}(x_i^{n-1}))] \tag{34}$$

Appendix B: Derivation of merging equations

The purpose of this appendix is to derive the update equations for the kth model variable. The derivation is similar to Poterjoy (2016), which we include here for the completeness and clarity of the paper. Assume that the kth model variable of the particle i is updated through

$$\begin{aligned}(x_{k,i})_{y_j}^u =&(\bar{x}_k)_{y_j} + r_{1,k}\left[(x_{k,s_i})_{y_{j-1}} - (\bar{x}_k)_{y_j}\right] \\ &+ r_{2,k}\left[(x_{k,t_i})_{y_{j-1}} - (\bar{x}_k)_{y_j}\right]\end{aligned} \tag{35}$$

Let the mean of the updated model variable be equal to the weighted mean of the proposal particles.

In equation (36), the mean of the resampled particles is approximated by the weighted mean of the particles before sampling. $(\tilde{\bar{x}}_k)_{y_j}$ is the mean of particles with the scalar weights $(w_i)_{y_j}$, and $(\hat{\bar{x}}_k)_{y_{j-1}}$ represents the mean of particles with the proposal weight w_i^*. Then,

$$r_{2,k} = -\frac{(\tilde{\bar{x}}_k)_{y_j} - (\bar{x}_k)_{y_j}}{(\hat{\bar{x}}_k)_{y_{j-1}} - (\bar{x}_k)_{y_j}} r_{1,k} = d_k r_{1,k} \tag{39}$$

Letting the variance of the updated model variable equal the weighted variance of the prior particles, we obtain equation (37). Then, the positive solution can be derived as in equation (38). For ease of calculation, the expression of d_k can be simplified.

$$\begin{aligned}d_k &= -\frac{(\tilde{\bar{x}}_k)_{y_j} - (\bar{x}_k)_{y_j}}{(\hat{\bar{x}}_k)_{y_{j-1}} - (\bar{x}_k)_{y_j}} \\ &= \frac{\sum_{i=1}^N \frac{(\omega_{k,i})_{y_j}}{(\Omega_k)_{y_j}}(x_{k,i})_{y_{j-1}} - \sum_{i=1}^N \frac{(w_i)_{y_j}}{(W)_{y_j}}(x_{k,i})_{y_{j-1}}}{\sum_{i=1}^N \frac{w_i^*}{U}(x_{k,i})_{y_{j-1}} - \sum_{i=1}^N \frac{(\omega_{k,i})_{y_j}}{(\Omega_k)_{y_j}}(x_{k,i})_{y_{j-1}}} \\ &= \frac{\sum_{i=1}^N \left[\frac{(\omega_{k,i})_{y_j}}{(\Omega_k)_{y_j}} - \frac{(w_i)_{y_j}}{(W)_{y_j}}\right](x_{k,i})_{y_{j-1}}}{\sum_{i=1}^N \left[\frac{w_i^*}{U} - \frac{(\omega_{k,i})_{y_j}}{(\Omega_k)_{y_j}}\right](x_{k,i})_{y_{j-1}}}\end{aligned} \tag{40}$$

If $\left[\frac{(\omega_{k,i})_{y_j}}{(\Omega_k)_{y_j}} - \frac{(w_i)_{y_j}}{(W)_{y_j}}\right]\left[\frac{1}{N} - \frac{(\omega_{k,i})_{y_j}}{(\Omega_k)_{y_j}}\right]^{-1}$ is constant for all particles, then d_k is a constant for all i. To determine it, let scalar weight $p_i = p(y_j|(x_i)_{y_{j-1}})$ and $P = \sum_{i=1}^N p_i w_i^*$, and the weights in (40) are

$$(\omega_{k,i})_{y_j} = [(p_i - 1)l_{j,k}^c + 1]w_i^* \tag{41}$$

$$(\Omega_k)_{y_j} = \sum_{i=1}^N (\omega_{k,i})_{y_j} \tag{42}$$

$$(w_i)_{y_j} = [(p_i - 1)\alpha + 1]w_i^* \tag{43}$$

$$(W)_{y_j} = \sum_{i=1}^N (w_i)_{y_j} \tag{44}$$

The vector total weight equation (41) is slightly different from equation (14). As long as the posterior particles can represent the posterior weights after each observation is assimilated, these two equations provide equivalent results. Then,

$$\begin{aligned}&\frac{\frac{(\omega_{k,i})_{y_j}}{(\Omega_k)_{y_j}} - \frac{(w_i)_{y_j}}{(W)_{y_j}}}{\frac{1}{N} - \frac{(\omega_{k,i})_{y_j}}{(\Omega_k)_{y_j}}} \\ &= \frac{\frac{l_{j,k}^c p_i w_i^* - l_{j,k}^c w_i^* + w_i^*}{l_{j,k}^c P - l_{j,k}^c U + U} - \frac{\alpha p_i w_i^* - \alpha w_i^* + w_i^*}{\alpha P - \alpha U + U}}{\frac{w_i^*}{U} - \frac{l_{j,k}^c p_i w_i^* - l_{j,k}^c w_i^* + w_i^*}{l_{j,k}^c P - l_{j,k}^c U + U}} \\ &= \frac{(\alpha - l_{j,k}^c)U}{l_{j,k}^c (W)_{y_j}}\end{aligned} \tag{45}$$

So $d_k = \frac{(\alpha - l_{j,k}^c)U}{l_{j,k}^c (W)_{y_j}}$ is constant for all particles, and depends on the parameter α, the localization function, the sum of the proposal weights and the sum of the total weights for the current observation.

$$
\begin{aligned}
(\bar{x}_k)_{y_j} &= (\bar{x}_k)^u_{y_j} \\
&= \frac{1}{N}\sum_{i=1}^{N}\left\{(\bar{x}_k)_{y_j} + r_{1,k}\left[(x_{k,s_i})_{y_{j-1}} - (\bar{x}_k)_{y_j}\right] + r_{2,k}\left[(x_{k,t_i})_{y_{j-1}} - (\bar{x}_k)_{y_j}\right]\right\} \\
&= (\bar{x}_k)_{y_j} + r_{1,k}\left[\frac{1}{N}\sum_{i=1}^{N}(x_{k,s_i})_{y_{j-1}} - (\bar{x}_k)_{y_j}\right] + r_{2,k}\left[\sum_{i=1}^{N}(x_{k,t_i})_{y_{j-1}} - (\bar{x}_k)_{y_j}\right] \\
&\approx (\bar{x}_k)_{y_j} + r_{1,k}\left[(\hat{\bar{x}}_k)_{y_j} - (\bar{x}_k)_{y_j}\right] + r_{2,k}\left[(\hat{\bar{x}}_k)_{y_{j-1}} - (\bar{x}_k)_{y_j}\right] \qquad (36) \\
(\sigma_k)^2_{y_j} &= var[(x_{k,i})^u_{y_j}] \\
&= \frac{1}{N-1}\sum_{i=1}^{N}\left\{(\bar{x}_k)_{y_j} + r_{1,k}\left[(x_{k,s_i})_{y_{j-1}} - (\bar{x}_k)_{y_j}\right] + r_{2,k}\left[(x_{k,t_i})_{y_{j-1}} - (\bar{x}_k)_{y_j}\right] - (\bar{x}_k)_{y_j}\right\}^2 \\
&= \frac{r^2_{1,k}}{N-1}\sum_{i=1}^{N}\left\{(x_{k,s_i})_{y_{j-1}} - (\bar{x}_k)_{y_j} + d_k\left[(x_{k,t_i})_{y_{j-1}} - (\bar{x}_k)_{y_j}\right]\right\}^2 \qquad (37)
\end{aligned}
$$

$$
r_{1,k} = \sqrt{\frac{(\sigma_k)^2_{y_j}(N-1)}{\sum_{i=1}^{N}\left\{(x_{k,s_i})_{y_{j-1}} - (\bar{x}_k)_{y_j} + d_k\left[(x_{k,t_i})_{y_{j-1}} - (\bar{x}_k)_{y_j}\right]\right\}^2}} \qquad (38)
$$

Appendix C: Comparison of merging and resampling

As stated in section 2, the merging step can be replaced by resampling. LWEnKF-SU and LPF-SU stand for the LWEnKF and LPF using the adjustment-minimising SU sampling rather than the merging step, respectively. The procedure for LWEnKF-SU is as follows:

1. Perform steps 1, 2, 3 of the procedure of LWEnKF in Section 2.
2. For each scalar observation y_j, perform step 5 of the procedure of LWEnKF in Section 2, but step 5) is replaced as follows:

   5) For each model variable k in $(1, 2, ..., N_x)$, perform adjustment-minimising SU sampling according to the total weights $\frac{(\omega_{k,i})_{y_j}}{(\Omega_k)_{y_j}}$, $i = 1,\ 2,\ ...,\ N$.
3. Perform step 6 of the procedure of LWEnKF in Section 2.

The experimental set-ups in Section 4.1.2 is considered to compare LWEnKF-SU and LPF-SU with LWEnKF and LPF. For each observation type, the average RMSEs for each method are shown in Figure 10 as a function of ensemble size. For all types of observations and all particle sizes, the results of the merging method are better than the adjustment-minimising SU sampling method. However, the improvement of the merging step on LWEnKF is smaller than that on LPF. When the number of particles is small, the RMSE of LPF is significantly lower than that of LPF-SU. Compared with the nonlinear observation operators, the improvement by the merging step of LPF is more obvious in the case of the linear observation operator.

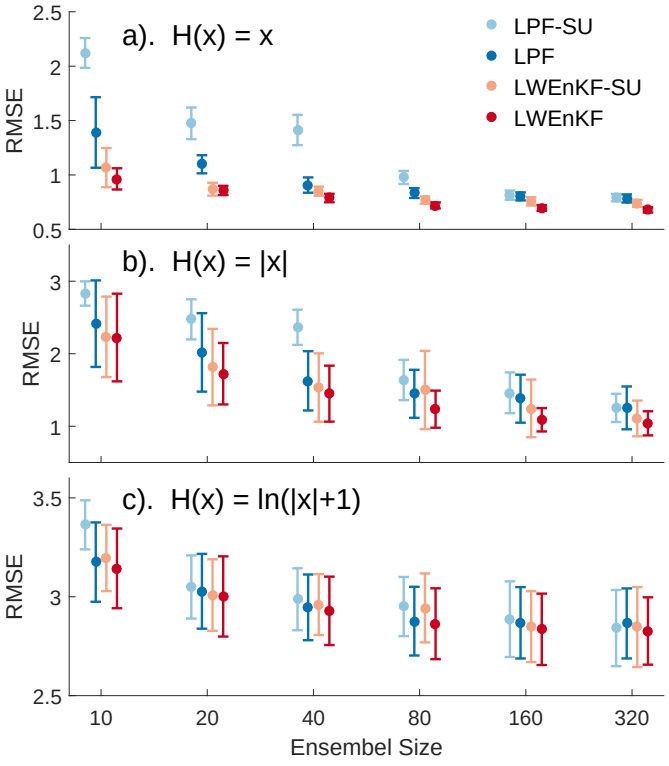

Figure 10. The average posterior RMSE over the data assimilation period using LWEnKF-SU, LPF-SU, LWEnKF and LPF as functions of ensemble size, in which the observation operators are a). $h(x) = x$, b). $h(x) = |x|$ and c). $h(x) = ln(|x| + 1)$.

Acknowledgement

The data used in the observation system simulation experiments are included within the paper. Thanks are extended to Dr. Zheqi Shen for providing the codes for the LPF with the L96 model used in this paper. The codes for the LEnKPF are modified according to https://github.com/robertsy/assimilr (Robert and Kunsch 2017).

This paper is supported by the National Key R&D Program of China (2018YFC1406202), the National Natural Science Foundation of China (NSFC, Grant No. 41675097), and the Hunan Provincial Innovation Foundation For Postgraduate (No. CX2017B034).

References

Anderson, J. L. (2001). An ensemble adjustment kalman filter for data assimilation. Monthly Weather Review, 129(12):2884–2903, doi:http://dx.doi.org/10.1175/1520-0493(2001)129

Anderson, J. L. (2003). A local least squares framework for ensemble filtering. Monthly Weather Review, 131(4), doi:http://dx.doi.org/10.1175/1520-0493(2003)131

Anderson, J. L. (2007). An adaptive covariance inflation error correction algorithm for ensemble filters. Tellus, 59A:210–224, doi:http://dx.doi.org/10.1111/j.1600-0870.2006.00216.x.

Bengtsson, T., Snyder, C., and Nychka, D. (2003). Toward a nonlinear ensemble filter for high-dimensional systems. Journal of Geophysical Research Atmospheres, 108(D24):–, doi:http://dx.doi.org/10.1029/2002JD002900.

Beyou, S., Cuzol, A., Gorthi, S. S., and Mémin, E. (2013). Weighted ensemble transform kalman filter for image assimilation. Tellus, 65(1):86–106, doi:http://dx.doi.org/10.3402/tellusa.v65i0.18803.

Bibov, A., Haario, H., and Solonen, A. (2015). Stabilized bfgs approximate kalman filter. Inverse Problems and Imaging, 9(4):1003–1024, doi:https://doi.org/10.3934/ipi.2015.9.1003.

Browne, P. A. and Van Leeuwen, P. J. (2015). Twin experiments with the equivalent weights particle filter and hadcm3. Quarterly Journal of the Royal Meteorological Society, 141(693):3399–3414, doi:http://dx.doi.org/10.1002/qj.2621.

Browne, P. A. (2016). A comparison of the equivalent weights particle filter and the local ensemble transform kalman filter in application to the barotropic vorticity equation. Tellus Series a-Dynamic Meteorology and Oceanography, 68, doi:10.3402/tellusa.v68.30466.

Burgers, G., Leeuwen, P. J. V., and Evensen, G. (1998). Analysis scheme in the ensemble kalman filter. Monthly Weather Review, 126:1719–1724, doi:https://dx.doi.org/10.1175/1520-0493(1998)126<1719:ASITEK>2.0.CO;2.

Evensen, G. (1994). Sequential data assimilation with a nonlinear quasi-geostrophic model using monte carlo methods to forecast error statistics. Journal of Geophysical Research: Oceans, 99C:10143–10162, doi:https://dx.doi.org/10.1029/94JC00572.

Farchi, A. and Bocquet, M. (2018). Review article: Comparison of local particle filters and new implementations. Nonlin. Processes Geophys, 25:765–807, doi:https://doi.org/10.5194/npg-25-765-2018.

Fisher, M., Trémolet, Y., Auvinen, H., Tan, D., and Poli, P. (2011). Weak-constraint and long window 4dvar. ECMWF Technical Memorandum, page 47.

Frei, M. and Künsch, H. R. (2013). Bridging the ensemble kalman and particle filters. Biometrika, 100(4):781–800, doi:http://dx.doi.org/10.1093/biomet/ast020.

Hamill, T. M. (2000). Interpretation of rank histograms for verifying ensemble forecasts. Monthly Weather Review, 129(3):550–560, doi:http://dx.doi.org/10.1175/1520-0493(2001)129

Houtekamer, P. L. and Mitchell, H. L. (1998). Data assimilation using an ensemble kalman filter technique. Mon.wea.rev, 126(3):796–811, doi:https://dx.doi.org/10.1175/1520-0493(1998)126<0796:DAUAEK>2.0.CO;2.

Houtekamer, P. and Mitchell, H. (2005). Ensemble kalman filtering. Quarterly Journal of the Royal Meteorological Society, 131(613):3269–3289, doi:http://dx.doi.org/10.1256/qj.05.135.

Lorenz, E. (2006). Predictability - a problem partly solved. in t. palmer & r. hagedorn (eds.), predictability of weather and climate (pp. 40-58). Cambridge: Cambridge University Press., doi:https://doi.org/10.1017/CBO9780511617652.004.

McGinnis, S., Nychka, D., and Mearns, L. O. (2015). A new distribution mapping technique for climate model bias correction. Machine Learning and Data Mining Approaches to Climate Science, V. Lakshmanan et al., Eds., Springer, 91–99, doi:http://dx.doi.org/10.1007/978-3-319-17220-0_9.

Morzfeld, M., Hodyss, D., and Snyder, C. (2016). What the collapse of the ensemble kalman filter tells us about particle filters. Statistics, doi:http://dx.doi.org/10.1080/16000870.2017.1283809.

Mussa, Z. and Kauranne, T. (2014). Data assimilation of two-dimensional geophysical flows with a variational ensemble kalman filter. Nonlinear Processes in Geophysics Discussions, 1(1):403–446, doi:https://doi.org/10.5194/npgd-1-403-2014.

Papadakis, N. (2007). Assimilation de donnees images: Applicationau suivi de courbes et de champs de vecteurs. ph.d. thesis, university of rennes i, 240 pp.

Papadakis, N., Mémin, E., Cuzol, A., and Gengembre, N. (2010). Data assimilation with the weighted ensemble kalman filter. Tellus A: Dynamic Meteorology and Oceanography, 62(5):673–697, doi:http://dx.doi.org/10.1111/j.1600-0870.2010.00461.x.

Pedlosky, J. (1987). Geophysical fluid dynamics, second edition. Springer, New York, NY, doi:https://doi.org/10.1007/978-1-4612-4650-3.

Penny, S. G. and Miyoshi, T. (2016). A local particle filter for high dimensional geophysical systems. Nonlinear Processes in Geophysics, 2(6):1631–1658, doi:http://dx.doi.org/10.5194/npg-23-391-2016.

Poterjoy, J. (2016). A localized particle filter for high-dimensional nonlinear systems. Monthly Weather Review, 144(1):59–76, doi:http://dx.doi.org/10.1175/MWR-D-15-0163.1.

Poterjoy, J. and Anderson, J. L. (2016). Efficient assimilation of simulated observations in a high-dimensional geophysical system using a localized particle filter. Monthly Weather Review, 144(5):2007–2020, doi:http://dx.doi.org/10.1175/mwr-d-15-0322.1.

Poterjoy, J., Sobash, R., and Anderson, J. (2017). Convective-scale data assimilation for the weather research and forecasting model using the local particle filter. Monthly Weather Review, 145(5):1897–1918, doi:http://dx.doi.org/10.1175/MWR-D-16-0298.1.

Poterjoy, J., Wicker, L., and Buehner, M. (2019). Progress toward the application of a localized particle filter for numerical weather prediction. Monthly Weather Review, 147(4):1107–1126, doi:http://10.1175/mwr-d-17-0344.1.

Robert, S. and Kunsch, H. R. (2017). Localizing the ensemble kalman particle filter. Tellus Series a-Dynamic Meteorology and Oceanography, 69, doi:10.1080/16000870.2017.1282016.

Robert, S., Leuenberger, D., and Künsch, H. R. (2017). A local ensemble transform kalman particle filter for convective scale data assimilation. Quarterly Journal of the Royal Meteorological Society., doi:http://dx.doi.org/10.1002/qj.3116.

Shen, Z. and Tang, Y. (2015). A modified ensemble kalman particle filter for non-gaussian systems with nonlinear measurement functions. Journal of Advances in Modeling Earth Systems, 7(1):50–66, doi:http://dx.doi.org/10.1002/2014MS000373.

Shen, Z., Tang, Y., and Li, X. (2017). A new formulation of vector weights in localized particle filters. Quarterly Journal of the Royal Meteorological Society, 143(709), doi:http://dx.doi.org/10.1002/qj.3180.

Snyder, C., Bengtsson, T., Bickel, P., and Anderson, J. (2008). Obstacles to high-dimensional particle filtering. Monthly Weather Review, 136(12):4629–4640, doi:http://dx.doi.org/10.1175/2008MWR2529.1.

Snyder, C., Bengtsson, T., and Morzfeld, M. (2015). Performance bounds for particle filters using the optimal proposal. Monthly Weather Review, 143(11):4750–4761, doi:10.1175/mwr-d-15-0144.1.

Tippett, M. K., Anderson, J. L., Bishop, C. H., Hamill, T. M., and Whitaker, J. S. (2003). Ensemble square root filters. Monthly Weather Review, 131(7):1485–1490, doi:http://dx.doi.org/10.1175/1520-0493(2003)131<1485:esrf>2.0.co;2.

Van Leeuwen, P. J. (2003). Nonlinear ensemble data assimilation for the ocean. in: Recent developments in data assimilation for atmosphere and ocean, ecmwf seminar 8-12 september 2003, reading, united kingdom. pp. 265–286.

Van Leeuwen, P. J. (2009). Particle filtering in geophysical systems. Monthly Weather Review, 137(12):4089–4114, doi:http://dx.doi.org/10.1175/2009MWR2835.1.

Van Leeuwen, P. J., Yuan, C., and Reich, S. (2015). Nonlinear Data Assimilation. Springer International Publishing.

Zhu, M., Van Leeuwen, P. J., and Amezcua, J. (2016). Implicit equal-weights particle filter. Quarterly Journal of the Royal Meteorological Society, 142(698):1904–1919, doi:http://dx.doi.org/10.1002/qj.2784.

