

RESEARCH ARTICLE

A localized weighted ensemble Kalman filter for high-dimensional systems

Yan Chen¹  | Weimin Zhang^{1,2} | Mengbin Zhu³¹College of Meteorology and Oceanology, National University of Defense Technology, Changsha, China²Laboratory of Software Engineering for Complex Systems, Changsha, China³Beijing Institute of Applied Meteorology, Beijing, China**Correspondence**

W. Zhang, College of Meteorology and Oceanology, National University of Defense Technology, Changsha 410073, China.

Email:wmzhang104@139.com

Funding information

Hunan Provincial Innovation Foundation For Postgraduate, No. CX2017B034; the National Natural Science Foundation of China, Grant No. 41675097

Abstract

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, the 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 the 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 the 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 (LEnKPF). Experiments showed that the LWEnKF has an obvious advantage over the LEnKPF when the number of particles is small, which indicates its potential for realistic applications limited by computing resources.

KEYWORDS

ensemble Kalman filter, localization, nonlinear observation operator, particle filter, proposal density

1 | INTRODUCTION

The ensemble Kalman filter (EnKF) and the particle filter (PF) are sequential data assimilation methods based on statistical theory. Compared with the PF, the 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 the EnKF actually limit the accuracy of analysis. The 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, the 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.

The PF has unique advantages in data assimilation, but there is a drawback. For the generalization of the ensemble idea, the 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 large, and the weights of most particles will be very small: 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 the PF in high-dimensional geophysical systems.

There are currently four main techniques for improving the PF to avoid filter degeneracy in practical applications (van Leeuwen *et al.* 2019). One technique is to introduce a proposal density that depends on the past model variables 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 the 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 the 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 the IEWPF has advantages over the 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 subsequent research, Beyou *et al.* (2013) used the ensemble transform Kalman filter as the proposal density. Although these two methods have the potential to combine the advantages of the 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). The localized WEnKF was discussed by van Leeuwen (2009), but the modifications in this paper were not considered.

Another technique to eliminate filter degeneracy is using a transformation process to move prior particles to posterior particles in a deterministic way. The ensemble transform PF (Reich, 2013) uses one transformation step, which fails in high-dimensional settings but allows for localization. The variational mapping PF, which pushes prior particles by using a sequence mapping that represents a particle flow, appears to have the potential for high-dimensional applications; van Leeuwen *et al.* (2019) give more information about transportation PFs.

The third way 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 the 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. The LPF expands the scalar weight of each particle to a vector one such that each model variable has a weight. For each observation, the 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).

Hybridisation is considered as a fourth way to mitigate filter degeneracy. Taking advantage of combining the characteristics of the EnKF and PF, Frei and Kunsch (2013) developed the ensemble Kalman particle filter (EnKPF), which was later modified by Shen and Tang (2015). Different from the idea of the WEnKF, the EnKPF divides assimilation into two stages with different proportions. The two stages adopt the 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 prediction (NWP) model and has assimilated real observations (Robert *et al.* 2017). Numerical experiments showed that the LEnKPF can improve the accuracy of the EnKF in strongly nonlinear systems. Combined with anamorphosis and regularization jitter, the numerical results of the LEnKPF also showed that in a linear system, it is comparable to the LETKF; Farchi and Bocquet (2018) give details. However, the computational cost of the adaptive choice of the proportion is relatively large.

In this paper, a similar localization approach to the LPF is applied to extend the WEnKF for high-dimensional systems. The localized EnKF is used to adjust the values 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 method was tested by applying it to the Lorenz 96 model and the quasi-geostrophic model. The obtained results indicate that localization can effectively solve the filter degeneracy problem and that the new filter performs similarly to or even better than the LPF or the EnKF in both linear and nonlinear cases.

In Section 2, we briefly introduce the idea of the WEnKF; then, we expand the scalar weights of particles to vectors and propose the new local weighted ensemble Kalman filter in Section 3. 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 spatially and 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 $q(\mathbf{x}^n | \mathbf{x}^{n-1}, \mathbf{y}^n)$, which is also called the proposal density. The posterior PDF of the model variable \mathbf{x}^n at time-step n given the observations \mathbf{y} at time-steps 1, 2, ..., n can be written as

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

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

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

\mathbf{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.

The weight is divided into two parts, with w_i^o and w_i^* denoting the likelihood weight and the proposal weight, respectively.

$$w_i^o = p(\mathbf{y}^n | \mathbf{x}_i^n) \quad (3)$$

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

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

$$w_i = w_i^0 \cdot w_i^*. \quad (5)$$

Considering the stochastic EnKF (Houtekamer and Mitchell, 2005) as the proposal density, Papadakis *et al.* (2010) proposed the weighted ensemble Kalman filter (WEnKF). The formulae 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).

$$r_{1,k} = \sqrt{\frac{(\sigma_k^j)^2}{\frac{1}{N-1} \sum_{i=1}^N \{x_{k,t_i}^{j-1} - \bar{x}_k^j + d_k[x_{k,t_i}^{j-1} - \bar{x}_k^j]\}^2}}. \quad (6)$$

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 the likelihood weight and propose the new local weighted ensemble Kalman filter (LWEnKF). If the observation errors are independent, that is, the covariance matrix \mathbf{R} is diagonal, then

$$w_i^0 = p(\mathbf{y}^n | \mathbf{x}_i^n) = \prod_{j=1}^{N_y} p(y_j^n | \mathbf{x}_i^n), \quad (7)$$

where y_j is the j th element of observation vector \mathbf{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 . Each element of ω is calculated through the formula as in Poterjoy *et al.* (2019)

$$\omega_{k,i} \propto \prod_{j=1}^{N_y} \left\{ \frac{(N\bar{w}_i^j - 1)l_{j,k}^c + 1}{N} \right\}, \quad (8)$$

where

$$\bar{w}_i^j = \frac{p(y_j | \mathbf{x}_i)}{\sum_{i=1}^N p(y_j | \mathbf{x}_i)}$$

and the subscript $k = 1, \dots, N_x$ is the model variable indicator. The localization function $l_{j,k}^c$ depends on the spatial location of observation y_j and model variable $\mathbf{x}_{k,i}$. The cut-off coefficient is denoted by c . We adopt the Gaspari–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 must be normalized before multiplication, which is different from the LPF.

Similar to Poterjoy (2016), a parameter $0 < \alpha < 1$ is used to adjust the scalar proposal weights (Equation (4)) and the scalar likelihood weights (Equation (10)). 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) and Poterjoy *et al.* (2019). We also use this strategy and make appropriate changes to fit the LWEnKF. First, a stochastic universal sampling is performed according to the proposal weights to obtain proposal-resampled particles. This resampling is performed to consider the influence of the proposal density on the weights. Then, for each observation y_j , the scalar total weights of all 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 non-parametric 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), Farchi and Bocquet (2018) and Poterjoy *et al.* (2019), 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 the LWEnKF.

In conclusion, when assimilating observation \mathbf{y} , the procedure for the LWEnKF is as follows:

1. Use the forecast model to evolve all the particles to the observation time step.
2. Perform the local perturbed EnKF analysis on each prior particle to obtain the proposal particles \mathbf{x}_i^0 , $i = 1, 2, \dots, N$.
3. Calculate the proposal weights w_i^* , $i = 1, 2, \dots, N$ (Equations (4), (A11) and (A12)). Adjust the proposal weights by the parameter α through

$$w_i^* = (w_i^* - 1)\alpha + 1 \quad (9)$$

and normalize it $w_i^* = w_i^* / \sum_{i=1}^N w_i^*$.

4. Assume that the scalar observation y_{j-1} is already assimilated. To consider the impact of the proposal weights, the total weight without localization for the updated model variable $x_{k,i}^{j-1}$ is saved in $w_{k,i}^{j-1}$ for each model variable k and each ensemble i . Clearly, $w_{k,i}^0 = w_i^*$ for all k . Then, observation y_j is processed as follows:

- a. For $i = 1, 2, \dots, N$, calculate the likelihood weight according to y_j and normalize it:

$$w_i^{oj} = \frac{[p(y_j | x_i^{j-1}) - 1]\alpha + 1}{\sum_{i=1}^N [p(y_j | x_i^{j-1}) - 1]\alpha + 1}. \quad (10)$$

- b. For each k in $\{k | l_{j,k} > 0\}$, calculate the total weights without localization for $x_{k,i}^{j-1}$:

$$w_{k,i}^j = w_i^{oj} * w_{k,i}^{j-1} \quad (11)$$

- c. For each model variable k in $\{k | l_{j,k} > 0\}$ and each particle i in $(1, 2, \dots, N)$, use proposal particles x_i^0 in Equation (8) to update the vector likelihood weights $\omega_{k,i}^j$ and derive the vector total weights

$$v_{k,i}^j = \omega_{k,i}^j \cdot w_i^* \quad (12)$$

The normalization factor is $\Omega_k^j = \sum_{i=1}^N v_{k,i}^j$.

- d. For each k in $\{k | l_{j,k} > 0\}$ and each i in $(1, 2, \dots, N)$, update the model variables by merging the particles via:

$$x_{k,i}^j = \bar{x}_k^j + r_{1,k} [x_{k,s_i}^{j-1} - \bar{x}_k^{j-1}] + r_{2,k} [x_{k,t_i}^{j-1} - \bar{x}_k^{j-1}], \quad (13)$$

where s_i is the resampled indicator for the total weights $w_{k,i}^j$, and t_i is the resampled indicator for the last step total weights $w_{k,i}^{j-1}$. The weighted average of the proposal particles is

$$\bar{x}_k^j = \sum_{i=1}^N \frac{v_{k,i}^j}{\Omega_k^j} x_{k,i}^0.$$

The parameters $r_{1,k}$ and $r_{2,k}$ provide the linear combination of the total-resampled particles and the proposal-resampled particles (Equations (6)), (14)–(16)):

$$d_k = \frac{N(1 - l_{j,k}^c)}{\tilde{\Omega}_k^j l_{j,k}^c}, \quad (14)$$

$$(\sigma_k^j)^2 = \sum_{i=1}^N \frac{v_{k,i}^j}{\Omega_k^j} [x_{k,i}^0 - \bar{x}_k^j]^2, \quad (15)$$

$$r_{2,k} = d_k r_{1,k}. \quad (16)$$

- e. After updating the particles x_k^j , the total weights also need to be updated for the next cycle through $w_{k,i}^j = 1/N$ for k in $\{k | l_{j,k} > 0\}$ and i in $(1, 2, \dots, N)$.
5. After all the observations at time-step n are assimilated sequentially, the updated particles obtained in the previous step are $\mathbf{x}_i^{N_y}$. The KDDM step is performed for each model variable $\mathbf{x}_k^{N_y}$ as follows:
 - a. Use Gaussian kernels to approximate the prior and posterior densities by particles with equal weight and the total vector weights.

$$p^p(x_k) = \frac{1}{N} \sum_{i=1}^N N(x_{k,i}^{N_y}, \sigma_k^{N_y}), \quad (17)$$

$$p^u(x_k) = \frac{v_{k,i}}{\Omega_k} \sum_{i=1}^N N(x_{k,i}^0, \sigma_k^0). \quad (18)$$

The kernel bandwidth is chosen as the sample standard deviation of the particles.

- b Use the trapezoid rule to form the cumulative distribution functions (cdfs) for prior (cdf_k^p) and posterior (cdf_k^u) cdfs.
- c Use cubic spline interpolation to find the cdf values of each proposal particle $c_{k,i}^p = \text{cdf}_k^p(x_{k,i}^{N_y})$.
- d Use cubic spline interpolation to adjust posterior particles $x_{k,i}^a = (\text{cdf}_k^u)^{-1}(c_{k,i}^p)$.

The derivation of Equations (6), (14)–(16) is similar to that in Poterjoy (2016), with only minor changes (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 $d_k \rightarrow 0$; thus,

$$\lim_{l_{j,k}^c \rightarrow 1} r_{1,k} = \sqrt{\frac{(\sigma_k^j)^2}{\frac{1}{N-1} \sum_{i=1}^N [x_{k,s_i}^{j-1} - \bar{x}_k^j]^2}} \approx 1. \quad (19)$$

The approximately equal sign in 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 $\lim_{l_{j,k}^c \rightarrow 1} r_{2,k} = 0$. 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 (13). Likewise, when the distance between the model variables and the observations is very large ($> 2c$), we derived $l_{j,k}^c = 0$ and $d_k \rightarrow \infty$, which leads to

$$\lim_{l_{j,k}^c \rightarrow 0} r_{1,k} = 0, \quad (20)$$

$$\lim_{l_{j,k}^c \rightarrow 0} r_{2,k} = \sqrt{\frac{(\sigma_k^j)^2}{\frac{1}{N-1} \sum_{i=1}^N [x_{k,t_i}^{j-1} - \bar{x}_k^j]^2}} = 1. \quad (21)$$

When $l_{j,k}^c \rightarrow 0$, 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 close to the observation location is updated by WEnKF, and the model variable far from the observation position is updated by the proposal. However, outside these domains, only the first two moments are considered by the merging strategy, so the KDDM is used additionally to adapt the particles to the higher-order moments.

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. The LWEnKF and LPF can still work without using the merging step but using resampling instead. Actually, in this case, the observations could be assimilated at once, which would be equivalent to the state-block localization described by Farchi and Bocquet (2018). To compare the merging step and the adjustment-minimizing 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 merging step. Therefore, this does not prevent the LWEnKF from being a nonlinear non-Gaussian filter.

4 | NUMERICAL EXPERIMENTS

In this section, several observation system simulation experiments are used to test the 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 the 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 the PFs and LEnKF. Then, nonlinearity and non-Gaussianity are introduced, which makes the data assimilation problem difficult for the LEnKF, where PFs excel. We also focus on the improvement of the LWEnKF and LPF by the KDDM, and the experiments are conducted in a 100-dimensional L96 model.

As a further comparison, the 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 the LWEnKF the observation error matrix \mathbf{R} is inflated by the inflation coefficient, which is chosen experimentally. The parameter α in the LPF and LWEnKF is chosen among 0.70–1.00 experimentally.

Experiments using the L96 model are programmed in the MATLAB language, in which the inflation of

background-error covariance in the 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 the 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 positions are evenly spaced on a circle. The evolution of each model variable over time is controlled by the following differential equations

$$\frac{dx_k}{dt} = (x_{k+1} - x_{k-2})x_{k-1} - x_k + F, \quad (22)$$

where $k = 1, 2, \dots, N_x$ and $x_{N_x+k} = x_k$, $x_{-k} = x_{N_x-k}$. To ensure chaos in the 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 $N(0, \mathbf{R})$ 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 $h = \mathbf{I}$. This configuration provides an advantage to the EnKF but would be challenging for PFs. The LWEnKF, LPF, LEnKF and LEnKPF are tested with ensemble sizes of 10, 20, 40, 80, 160, and 320.

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. The LWEnKF is always significantly superior to the LPF. The LWEnKF is superior to the LEnKPF when the ensemble size is relatively small ($N = 10, 20$) and is comparable to the LEnKF when the ensemble size reaches 320. When the ensemble size exceeds 160, the RMSEs obtained by the LEnKPF are lower than those obtained by the other three filters. The LPF obtains higher RMSEs than the other three

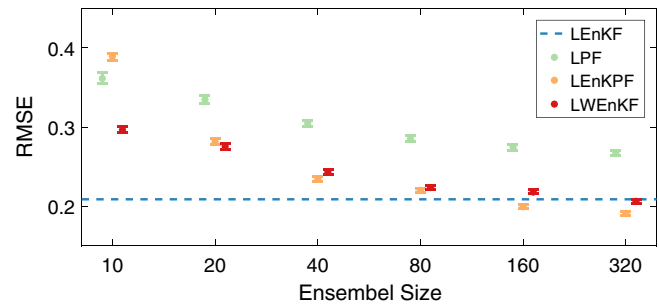


FIGURE 1 The average posterior RMSE over the data assimilation period using the LWEnKF, LPF, LEnKF, and LEnKPF as a function of ensemble size, in which the observation operator is $h(x) = x$

filters except for that of the 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 the 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 $h(x) = x$, while the last two sets used nonlinear functions $h(x) = |x|$ and $h(x) = \ln(|x| + 1)$, in which the latter has more nonlinearity than the former.

In this test, nonlinearity and non-Gaussianity are introduced in several ways: using rare and sparse observations, using nonlinear observation operators, and using non-Gaussian model errors. These set-ups make the data assimilation problems easier for PFs than for the 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 the LWEnKF for ensemble sizes as small as 10. That is,

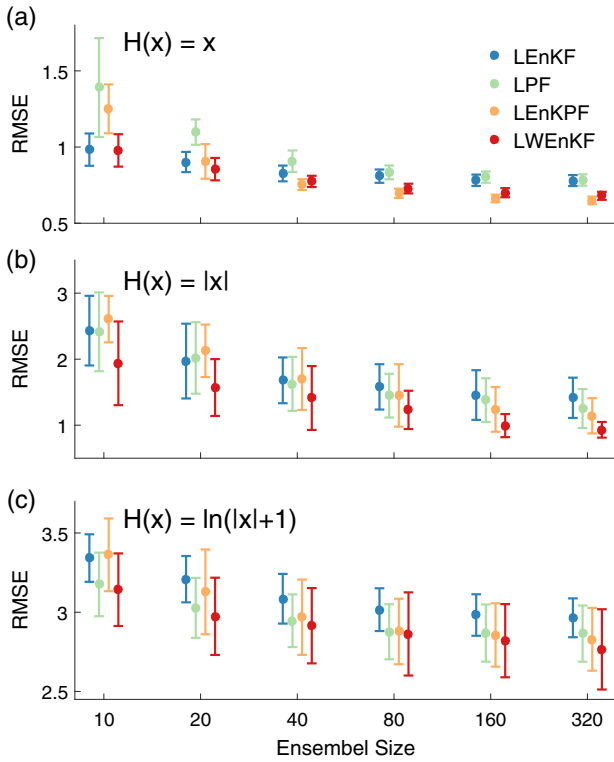


FIGURE 2 The average posterior RMSE over the data assimilation period using the LWEnKF, LPF, LEnKF, and LEnKPF as a function of ensemble size, in which the observation operators are (a) $h(x) = x$, (b) $h(x) = |x|$ and (c) $h(x) = \ln(|x| + 1)$

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

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

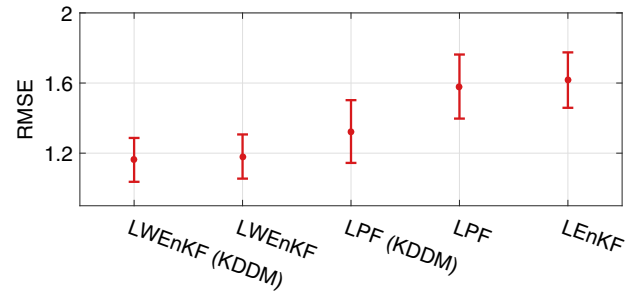


FIGURE 3 The average posterior RMSE over the data assimilation period for the different methods

nonlinear data. With both nonlinear observation operators, the RMSEs obtained by the LEnKPF are higher than those of the LWEnKF.

For both linear and nonlinear observation operators cases, the LWEnKF has an obvious advantage over the other filters when the number of particles is small ($N = 10, 20$). For more than 40 particles, the LEnKPF is slightly better than the LWEnKF for the case of the linear observation operator, and for the case of the nonlinear observation operators the LEnKPF is worse than the LWEnKF. The LEnKF and LPF have shortcomings in the case of the nonlinear and linear observation operators, respectively, and the LWEnKF is no worse than these two methods under all three observation operator cases. In this sense, we believe that the proposed method can effectively promote the advantages of the 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 the LWEnKF, we increase the dimension of the L96 model to a relatively high 100 dimensions. We chose the 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 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 the KDDM was applied, both the LWEnKF and LPF resulted in a lower RMSE than those without the KDDM. The LWEnKF using no KDDM is even more outstanding than the LPF with the KDDM. Comparison of the ratio of the average RMSE to the average spread illustrated in Figure 4 indicates that the spread of the LEnKF is too low. When no KDDM step is used, the LPF and LWEnKF perform with too much and too little spread, respectively. Applying the KDDM

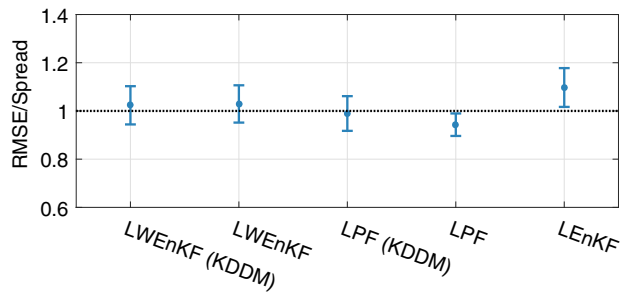


FIGURE 4 The ratio of the average posterior RMSE to the average ensemble spread for the different methods

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 fortieth model variable from the LEnKF, LPF and LWEnKF. The abscissae of these

figures are 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 the LEnKF indicate strong underdispersion. The histograms of the LWEnKF without the KDDM are slightly sunken, which can be slightly alleviated by the KDDM. The humped histograms of the LPF show overdispersion, which can be modified into a relatively uniform distribution by the KDDM. Some of the histograms have high values at both ends, which means that the truth falls outside the range of the particles. This also occurred 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

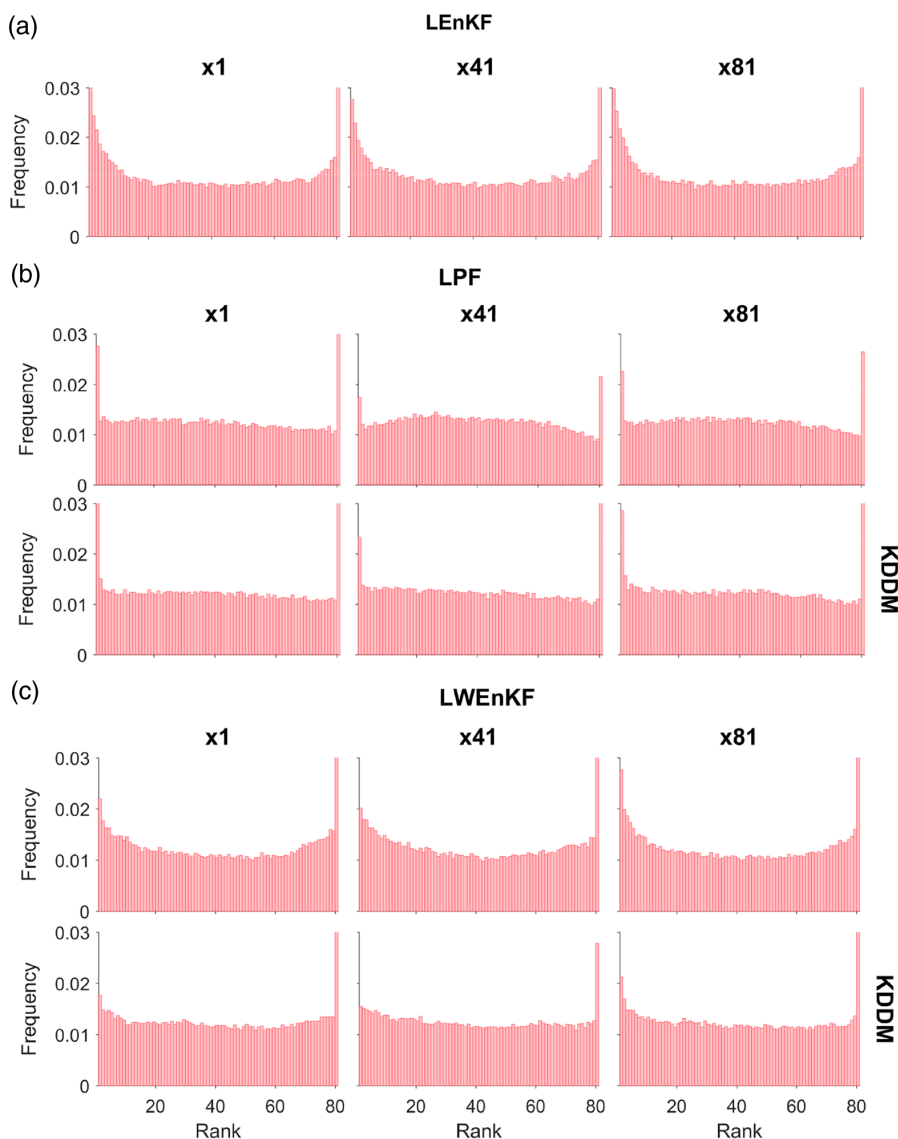


FIGURE 5 The rank histograms calculated from posterior particles generated by (a) LEnKF, (b) LPF without (first row) or with (second row) KDDM and (c) the LWEnKF without (first row) or with (second row) KDDM. The columns represent the model variables 1, 41 and 81

of the 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 the KDDM are far smaller than those of the other experiments for almost all the variables at almost all time steps, which strongly indicates that the LWEnKF can significantly outperform the LEnKF and LPF with affordable particles. In addition, even if the KDDM is not adopted, the analysis errors of the LWEnKF appear to be smaller than those of the LEnKF and LPF with the KDDM, which further indicates that the new filter is better than the other two filters 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 method can combine the advantages of the LEnKF for the linear observation operator and the 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 (Fisher *et al.* 2011; Mussa *et al.* 2014; Bibov *et al.* 2015).

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

$$Q = \begin{bmatrix} \sigma_u^2 I & & & \\ & \sigma_v^2 I & & \\ & & \sigma_\psi^2 I & \\ & & & \sigma_q^2 I \end{bmatrix}. \quad (23)$$

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 the 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 to 1000 days of the assimilation period. The LPF and LWEnKF outperform the LEnKF in the experiments using logarithmic observation operators, which demonstrates the advantages of the particle filters. The 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 the LPF is higher than that of the LEnKF, which is reasonable due to the small number of particles, and also reflects the shortcomings of the LPF. However, the 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 stream function, the performance of the LWEnKF is comparable to that of the wind

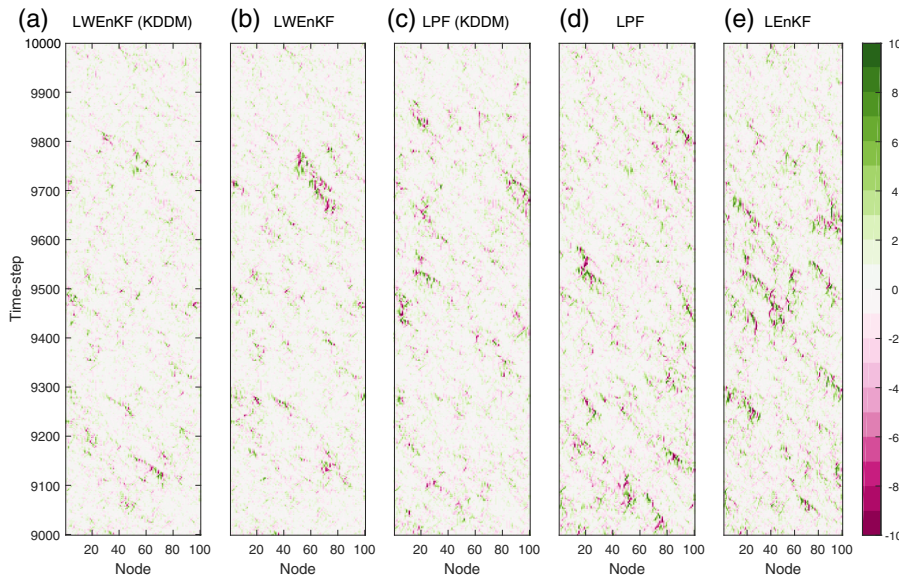


FIGURE 6 The posterior errors for the last 1000 time steps using (a) LEnKF, (b) LPF without KDDM, (c) LPF with KDDM, (d) LEnKF without KDDM and (e) LPF with KDDM

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

	$\mathbf{h(x) = x}$	$ \mathbf{x} $	$\log(\mathbf{x} - \bar{\mathbf{x}})$
	RMSE, Spread		
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	0.3956, 0.2935	0.4739, 0.3461	1.5883, 0.8105
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	2.9876, 1.5588	3.2256, 1.7881	5.3193, 2.9208
Zonal wind u ($\text{m}\cdot\text{s}^{-1}$)			
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	0.3966, 0.2973	0.4661, 0.3434	1.2279, 0.6506
Meridional wind v ($\text{m}\cdot\text{s}^{-1}$)			
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	0.3497, 0.2560	0.4145, 0.3011	1.0509, 0.5634

components, which indicates that, for multivariate models with complex relationships, the LWEnKF can sample in the high-probability region of the posterior PDF.

The following presents some results for experiments using the nonlinear observation operator $h(x) = |x|$.

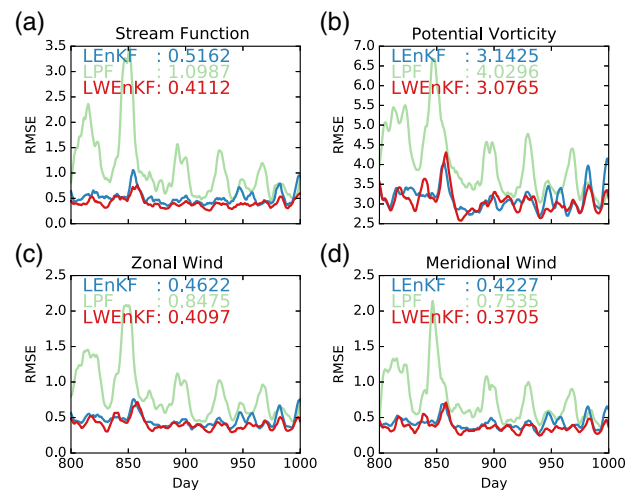


FIGURE 7 Analysis RMSE of the LEnKF (blue), LPF (green) and LWEnKF (red) at every analysis time step for (a) stream function, (b) potential vorticity, (c) zonal wind and (d) meridional wind. The values in the upper left corner of each panel are the time-averaged RMSEs of the three filters

Figure 7 shows the RMSEs of the LEnKF, LPF and LWEnKF at each analysis time step, where the RMSE of the LWEnKF is the lowest for most time steps. Moreover, comparing the observed wind components with the unobserved potential vorticity and stream function, the trend of their RMSE time series is similar, which is consistent with the time-averaged RMSEs in Table 1.

The truth and comparison of the three filters' analyses in potential vorticity and stream function at 1000 day are illustrated in Figure 8. We subtract the truth from the analyses in Figure 8 to obtain Figure 9 where it can be clearly

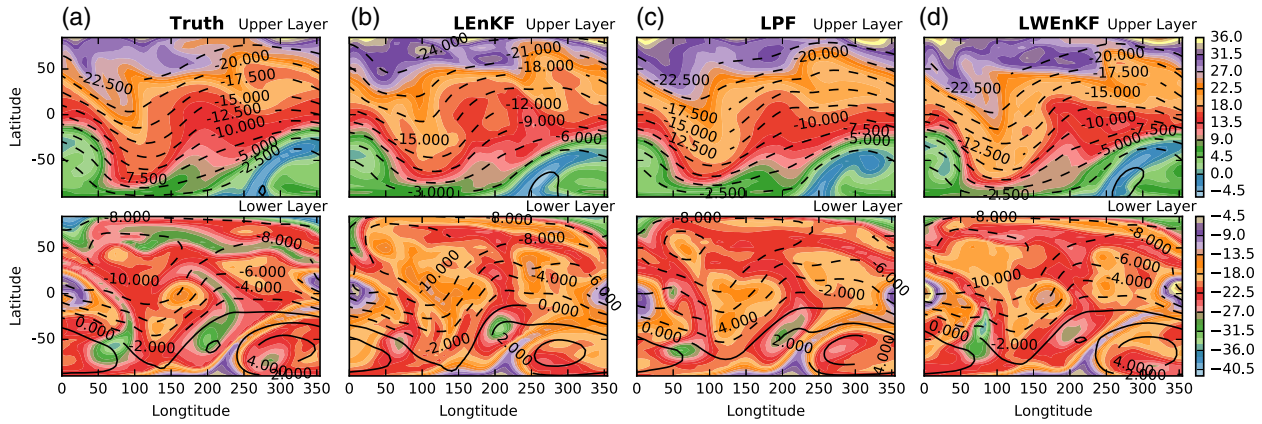


FIGURE 8 The (a) truth and analyses of the (b) LEnKF, (c) LPF and (d) LWEnKF for potential vorticity (shading) and stream function (contours, solid for positive values and dashed for negative values) at 1000 day

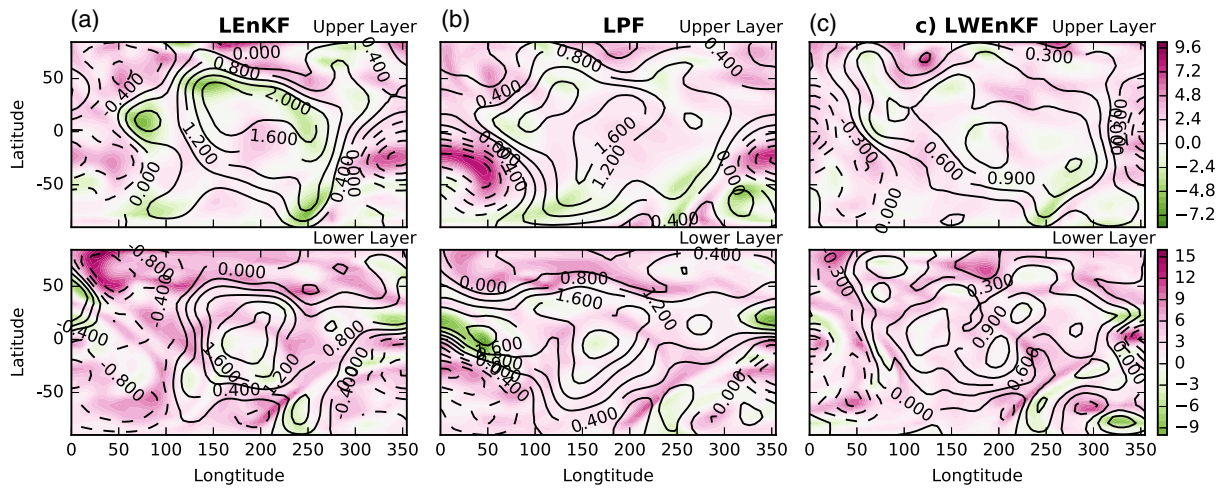


FIGURE 9 The departures between the truth and analyses of the (a) LEnKF, (b) LPF and (c) LWEnKF for potential vorticity (shading) and stream function (contours, solid for positive values and dotted for negative values) at 1000 day for both upper and lower layers

seen that the analysis field of the LWEnKF is the closest to the truth among the three data assimilation methods.

5 | CONCLUSIONS

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

The LWEnKF first uses the 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 merging steps 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 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 from the observations, the particles only retain information from the LEnKF and the proposal weights.

Using the L96 model, we tested the effectiveness of the LWEnKF and compared it with that of the LPF, LEnKF and LEnKPF. In the experiments with a mildly nonlinear configuration, the LWEnKF is always significantly superior to the LPF and is superior to the LEnKPF when the ensemble size is relatively small. However, the

LWEnKF needs a large ensemble size to be comparable to the LEnKF. The LEnKPF shows significant advantages over the 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 the LEnKF are more efficient than LPF when using the linear observation operator. 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, the LPF can provide better results than the LEnKF when acceptable particles are adopted, which is consistent with the conclusions in Poterjoy (2016). In both the linear and nonlinear observation operator cases, the LWEnKF can combine the advantages of the LEnKF and LPF and achieve similar or better assimilation efficiency. When the number of particles is small, the LWEnKF has an obvious advantage over the LEnKPF, LEnKF and LPF. This indicates the potential of the LWEnKF for realistic applications since atmospheric or oceanic data assimilation systems can use only 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 the 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 as in the LPF and LEnKF, the analysis fields obtained by the LWEnKF are the closest to the truth. This result reflects the efficiency of the new approach and its potential for application in NWP models.

The 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 proposal weights are calculated using the model error covariance matrix, which may be highly degenerate in a realistic geophysical model. Although the weight inflation through α can mitigate the degeneracy, this is an artificial technique. If using $\alpha = 0$, the proposal weights and the total weights without localization of the LWEnKF are equal weights, which causes the particles to represent PDF incorrectly. To eliminate this degeneracy, the proposal weights could also be considered localized, and relevant research is under way.

ACKNOWLEDGEMENTS

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 here. The codes for the LEnKPF are modified according to <https://github.com/robertsy/assimilr> (accessed 31 October 2019; 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).

ORCID

Yan Chen  <https://orcid.org/0000-0003-2066-6064>

REFERENCES

- Anderson, J.L. (2001) An ensemble adjustment Kalman filter for data assimilation. *Monthly Weather Review*, 129(12), 2884–2903
- Anderson, J.L. (2003) A local least squares framework for ensemble filtering. *Monthly Weather Review*, 131(4). [https://doi.org/10.1175/1520-0493\(2003\)131<3C0634:ALLSFF>3E2.0.CO;2](https://doi.org/10.1175/1520-0493(2003)131<3C0634:ALLSFF>3E2.0.CO;2)
- Anderson, J.L. (2007) An adaptive covariance inflation error correction algorithm for ensemble filters. *Tellus A*, 59(2), 210–224. <https://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*, 108(D24). <https://doi.org/10.1029/2002JD002900>
- Beyou, S., Cuzol, A., Gorthi, S.S. and Mémén, E. (2013) Weighted ensemble transform Kalman filter for image assimilation. *Tellus A*, 65(1), 86–106. <https://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. <https://doi.org/10.3934/ipi.2015.9.1003>
- 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 A*, 68. <https://doi.org/10.3402/tellusa.v68.30466>
- 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, 3399–3414. <https://doi.org/10.1002/qj.2621>
- Burgers, G., van Leeuwen, P.J. and Evensen, G. (1998) Analysis scheme in the ensemble Kalman filter. *Monthly Weather Review*, 126, 1719–1724. [https://doi.org/10.1175/1520-0493\(1998\)126<1719:ASITEK>2.0.CO;2](https://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*, 99C, 10143–10162. <https://doi.org/10.1029/94JC00572>
- Farchi, A. and Bocquet, M. (2018) Review article: comparison of local particle filters and new implementations. *Nonlinear Processes in Geophysics*, 25, 765–807. <https://doi.org/10.5194/npg-25-765-2018>
- Fisher, M., Trémolet, Y., Auvinen, H., Tan, D.G.H. and Poli, P. (2011) *Weak-Constraint and Long-Window 4DVAR*. Technical Memorandum 655, ECMWF, Reading, UK.
- Frei, M. and Künsch, H.R. (2013) Bridging the ensemble Kalman and particle filters. *Biometrika*, 100(4), 781–800. <https://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. [https://doi.org/10.1175/1520-0493\(2001\)129<0550:IORHFV>2.0.CO;2](https://doi.org/10.1175/1520-0493(2001)129<0550:IORHFV>2.0.CO;2)
- Houtekamer, P.L. and Mitchell, H.L. (2005) Ensemble Kalman filtering. *Quarterly Journal of the Royal Meteorological Society*, 131, 3269–3289. <https://doi.org/10.1256/qj.05.135>
- Houtekamer, P.L. and Mitchell, H.L. (1998) Data assimilation using an ensemble Kalman filter technique. *Monthly Weather Review*, 126(3), 796–811. [https://doi.org/10.1175/1520-0493\(1998\)126<0796:DAUAEK>2.0.CO;2](https://doi.org/10.1175/1520-0493(1998)126<0796:DAUAEK>2.0.CO;2)
- Lorenz, E.N. (2006). Predictability: a problem partly solved, in Predictability of Weather and Climate. Cambridge University Press, Cambridge, UK.
- McGinnis, S., Nychka, D. and Mearns, L.O. (2015). A new distribution mapping technique for climate model bias correction, in Machine Learning and Data Mining Approaches to Climate Science. Lakshmanan, V., Gilleland, E., McGovern, A., Tingley, M. (eds), Springer, Berlin, pp. 91–99, 10.1007/978-3-319-17220-0_9, (to appear in print).
- Morzfeld, M., Hodyss, D. and Snyder, C. (2016) What the collapse of the ensemble Kalman filter tells us about particle filters. *Tellus A*, 69(1283809). <https://doi.org/10.1080/16000870.2017.1283809>
- Mussa, Z., Amour, I., Bibov, A. and Kauranne, T. (2014) Data assimilation of two-dimensional geophysical flows with a variational ensemble Kalman filter. *Nonlinear Processes in Geophysics Discussions*, 1, 403–446. <https://doi.org/10.5194/npgd-1-403-2014>
- Papadakis, N. (2007). Assimilation de données images: application au suivi de courbes et de champs de vecteurs. PhD thesis, Université de Rennes, France.
- Papadakis, N., Mémin, E., Cuzol, A. and Gengembre, N. (2010) Data assimilation with the weighted ensemble Kalman filter. *Tellus A*, 62(5), 673–697. <https://doi.org/10.1111/j.1600-0870.2010.00461.x>
- Pedlosky, J. (1987) *Geophysical Fluid Dynamics* (2nd ed.) Springer, New York, NY.
- Penny, S.G. and Miyoshi, T. (2016) A local particle filter for high-dimensional geophysical systems. *Nonlinear Processes in Geophysics*, 2(6), 1631–1658. <https://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. <https://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. <https://doi.org/10.1175/mwr-d-15-0322.1>
- Poterjoy, J., Sobash, R. and Anderson, J.L. (2017) Convective-scale data assimilation for the weather research and forecasting model using the local particle filter. *Monthly Weather Review*, 145(5), 1897–1918. <https://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. <https://doi.org/10.1175/mwr-d-17-0344.1>
- Reich, S. (2013) A nonparametric ensemble transform method for Bayesian inference. *SIAM Journal on Scientific Computing*, 35(4), A2013–A2024. <https://doi.org/10.1137/130907367>
- Robert, S. and Kunsch, H.R. (2017) Localizing the ensemble Kalman particle filter. *Tellus A*, 69. <https://doi.org/10.1080/16000870.2017.1282016>
- Robert, S., Leuenberger, D. and Kunsch, H.R. (2017) A local ensemble transform Kalman particle filter for convective-scale data assimilation. *Quarterly Journal of the Royal Meteorological Society*, 144, 1279–1296. <https://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. <https://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, 3269–3278. <https://doi.org/10.1002/qj.3180>
- Snyder, C., Bengtsson, T., Bickel, P. and Anderson, J.L. (2008) Obstacles to high-dimensional particle filtering. *Monthly Weather Review*, 136(12), 4629–4640. <https://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. <https://doi.org/10.1175/mwr-d-15-0144.1>
- van Leeuwen, P.J. (2003). Nonlinear ensemble data assimilation for the ocean. In: Seminar on recent developments in data assimilation for atmosphere and ocean, 8–12 September 2003. ECMWF, Reading, UK.
- van Leeuwen, P.J. (2009) Particle filtering in geophysical systems. *Monthly Weather Review*, 137(12), 4089–4114. <https://doi.org/10.1175/2009MWR2835.1>
- van Leeuwen, P.J., Yuan, C. and Reich, S. (2015) *Nonlinear Data Assimilation*. Springer, Berlin.
- van Leeuwen, P.J., Kunsch, H.R., Nerger, L., Potthast, R. and Reich, S. (2019) Particle filters for high-dimensional geoscience applications: a review. *Quarterly Journal of the Royal Meteorological Society*, 145, 2335–2365. <https://doi.org/10.1002/qj.3551>
- Zhu, M., van Leeuwen, P.J. and Amezcua, J. (2016) Implicit equal-weights particle filter. *Quarterly Journal of the Royal Meteorological Society*, 142, 1904–1919. <https://doi.org/10.1002/qj.2784>

How to cite this article: Chen Y, Zhang W, Zhu M. A localized weighted ensemble Kalman filter for high-dimensional systems. *Q.J.R. Meteorol. Soc.* 2019;1–16. <https://doi.org/10.1002/qj.3685>

APPENDICES

A. FORMULATION OF PROPOSAL WEIGHTS

Recall that the assumptions in Section 2 still hold. Assume the model equation is

$$\mathbf{x}_i^{n,f} = \mathbf{M}(\mathbf{x}_i^{n-1}) + \beta_i^n, \quad (\text{A1})$$

$$\beta_i^n \sim \mathbf{N}(0, \mathbf{Q}), \quad (\text{A2})$$

where $M(\cdot)$ is the nonlinear deterministic forecast model and the model error β is Gaussian with zero mean and covariance matrix \mathbf{Q} . The equations for the localized stochastic EnKF (Houtekamer and Mitchell, 2005) are

$$\epsilon_i \sim N(0, \mathbf{R}), \quad (\text{A3})$$

$$\mathbf{x}_i^n = \mathbf{x}_i^{n,f} + \mathbf{K}_e[\mathbf{y} + \epsilon_i - h(\mathbf{x}_i^{n,f})], \quad (\text{A4})$$

$$\mathbf{K}_e = l o(\mathbf{P}h^T)[l o(h\mathbf{P}h^T) + \mathbf{R}]^{-1}, \quad (\text{A5})$$

$$\mathbf{P}h^T = \frac{1}{N-1} \sum_{i=1}^N (\mathbf{x}_i^{n,f} - \overline{\mathbf{x}_i^{n,f}})[h(\mathbf{x}_i^{n,f}) - \overline{h(\mathbf{x}_i^{n,f})}]^T, \quad (\text{A6})$$

$$h\mathbf{P}h^T = \frac{1}{N-1} \sum_{i=1}^N [h(\mathbf{x}_i^{n,f}) - \overline{h(\mathbf{x}_i^{n,f})}][h(\mathbf{x}_i^{n,f}) - \overline{h(\mathbf{x}_i^{n,f})}]^T, \quad (\text{A7})$$

where $h(\cdot)$ is the nonlinear observation operator. The observation error is denoted as ϵ . The Kalman gain matrix \mathbf{K}_e is calculated from the ensemble and localized via localization function l .

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

$$\begin{aligned} h(\mathbf{x}_i^{n,f}) &= h\{M(\mathbf{x}_i^{n-1}) + \beta_i^n\} \\ &\approx h\{M(\mathbf{x}_i^{n-1})\} + \mathbf{H}\beta_i^n, \end{aligned} \quad (\text{A8})$$

where \mathbf{H} is the tangent linear observation operator corresponding to h . The right-hand side of Equation (A4) can be divided into two parts: the deterministic part and the remaining is the stochastic part. If the model error is not related to the observation error, the covariance matrix of the stochastic part is

$$\Sigma = (\mathbf{I} - \mathbf{K}_e\mathbf{H})\mathbf{Q}(\mathbf{I} - \mathbf{K}_e\mathbf{H})^T + \mathbf{K}_e\mathbf{R}\mathbf{K}_e^T. \quad (\text{A9})$$

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

$$q(\mathbf{x}_i^n | \mathbf{x}_i^{n-1}, \mathbf{y}^n) \propto \exp[(\mathbf{x}_i^n - \mu_i^n)^T \Sigma^{-1}(\mathbf{x}_i^n - \mu_i^n)]. \quad (\text{A10})$$

Finally, it is easy to derive the transition density related to the model Equations (A1) and (A2), which is also the numerator of the proposal weight:

$$p(\mathbf{x}_i^n | \mathbf{x}_i^{n-1}) \propto \exp[\{\mathbf{x}_i^n - M(\mathbf{x}_i^{n-1})\}^T \mathbf{Q}^{-1} \{\mathbf{x}_i^n - M(\mathbf{x}_i^{n-1})\}]. \quad (\text{A11})$$

B. DERIVATION OF MERGING EQUATIONS

The purpose of this appendix is to derive the update equations for the k th model variable. The derivation is similar to Poterjoy (2016) and Poterjoy *et al.* (2019), which

we include here for the completeness and clarity of the paper. Assume that the k th model variable of the particle i is updated through

$$(x_{k,i}^j)^u = \bar{x}_k^j + r_{1,k}[x_{k,s_i}^{j-1} - \bar{x}_k^j] + r_{2,k}[x_{k,t_i}^{j-1} - \bar{x}_k^j]. \quad (\text{B1})$$

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

$$\begin{aligned} \bar{x}_k^j &= \overline{(x_{k,i}^j)^u} \\ &= \frac{1}{N} \sum_{i=1}^N \{\bar{x}_k^j + r_{1,k}[x_{k,s_i}^{j-1} - \bar{x}_k^j] + r_{2,k}[x_{k,t_i}^{j-1} - \bar{x}_k^j]\} \\ &= \bar{x}_k^j + r_{1,k} \left[\frac{1}{N} \sum_{i=1}^N x_{k,s_i}^{j-1} - \bar{x}_k^j \right] + r_{2,k} \left[\frac{1}{N} \sum_{i=1}^N x_{k,t_i}^{j-1} - \bar{x}_k^j \right] \\ &\approx \bar{x}_k^j + r_{1,k}[\tilde{x}_k^j - \bar{x}_k^j] + r_{2,k}[\hat{x}_k^{j-1} - \bar{x}_k^j], \end{aligned} \quad (\text{B2})$$

$$\begin{aligned} (\sigma_k^j)^2 &= \text{var}[(x_{k,i}^j)^u] = \frac{1}{N-1} \sum_{i=1}^N \{\bar{x}_k^j + r_{1,k}[x_{k,s_i}^{j-1} - \bar{x}_k^j] \\ &\quad + r_{2,k}[x_{k,t_i}^{j-1} - \bar{x}_k^j] - \bar{x}_k^j\}^2 \\ &= \frac{r_{1,k}^2}{N-1} \sum_{i=1}^N \{x_{k,s_i}^{j-1} - \bar{x}_k^j + d_k[x_{k,t_i}^{j-1} - \bar{x}_k^j]\}^2, \end{aligned} \quad (\text{B3})$$

$$r_{1,k} = \sqrt{\frac{(\sigma_k^j)^2(N-1)}{\sum_{i=1}^N \{x_{k,s_i}^{j-1} - \bar{x}_k^j + d_k[x_{k,t_i}^{j-1} - \bar{x}_k^j]\}^2}}. \quad (\text{B4})$$

In Equation (B2), the mean of the updated particles is approximated by the weighted mean of the particles before sampling. \tilde{x}_k^j is the mean of particles with total weights $w_{k,i}^j$, and \hat{x}_k^{j-1} represents the mean of particles with total weights $w_{k,i}^{j-1}$. Then,

$$r_{2,k} = -\frac{\tilde{x}_k^j - \bar{x}_k^j}{\hat{x}_k^{j-1} - \bar{x}_k^j} r_{1,k} = d_k r_{1,k}. \quad (\text{B5})$$

Letting the variance of the updated model variable equal the weighted variance of the proposal particles, we obtain Equation (B3). Then, the positive solution can be derived as in Equation (B4). The updated particles can represent the proposal particles with vector total weights after each observation is assimilated:

$$\tilde{x}_k^j \approx \sum_{i=1}^N \frac{\tilde{w}_i^j v_{k,i}^{j-1}}{\tilde{\Omega}_k^j} x_{i,j}^0, \quad (\text{B6})$$

$$\hat{x}_k^{j-1} \approx \sum_{i=1}^N \frac{v_{k,i}^{j-1}}{\Omega_k^{j-1}} x_{i,j}^0, \quad (\text{B7})$$

where $\tilde{\Omega}_k^j$ normalizes $\tilde{w}_i^j v_{k,i}^{j-1}$. For ease of calculation, the expression of d_k can be simplified:

$$d_k = \frac{\bar{x}_k^j - \tilde{x}_k^j}{\bar{x}_k^{j-1} - \tilde{x}_k^j} = \frac{\sum_{i=1}^N \frac{v_{k,i}^j x_{i,j}^0}{\tilde{\Omega}_k^j} - \sum_{i=1}^N \frac{\tilde{w}_i^j v_{k,i}^{j-1} x_{i,j}^0}{\tilde{\Omega}_k^j}}{\sum_{i=1}^N \frac{v_{k,i}^{j-1} x_{i,j}^0}{\tilde{\Omega}_k^{j-1}} - \sum_{i=1}^N \frac{v_{k,i}^j x_{i,j}^0}{\tilde{\Omega}_k^j}} \quad (\text{B8})$$

$$= \frac{\sum_{i=1}^N \left[\frac{v_{k,i}^j}{\tilde{\Omega}_k^j} - \frac{\tilde{w}_i^j v_{k,i}^{j-1}}{\tilde{\Omega}_k^j} \right] x_{i,j}^0}{\sum_{i=1}^N \left[\frac{v_{k,i}^{j-1}}{\tilde{\Omega}_k^{j-1}} - \frac{v_{k,i}^j}{\tilde{\Omega}_k^j} \right] x_{i,j}^0}.$$

If $\left[\frac{v_{k,i}^j}{\tilde{\Omega}_k^j} - \frac{\tilde{w}_i^j v_{k,i}^{j-1}}{\tilde{\Omega}_k^j} \right] \left[\frac{v_{k,i}^{j-1}}{\tilde{\Omega}_k^{j-1}} - \frac{v_{k,i}^j}{\tilde{\Omega}_k^j} \right]^{-1}$ is constant for all particles, then d_k is a constant for all i . To determine the value, substitute

$$v_{k,i}^j = \frac{(N\tilde{w}_i^j - 1)l_{j,k}^c + 1}{N} v_{k,i}^{j-1},$$

then

$$\frac{\frac{v_{k,i}^j}{\tilde{\Omega}_k^j} - \frac{\tilde{w}_i^j v_{k,i}^{j-1}}{\tilde{\Omega}_k^j}}{\frac{v_{k,i}^{j-1}}{\tilde{\Omega}_k^{j-1}} - \frac{v_{k,i}^j}{\tilde{\Omega}_k^j}} = \frac{N(1 - l_{j,k}^c)}{\tilde{\Omega}_k^j l_{j,k}^c}. \quad (\text{B9})$$

So $d_k = N(1 - l_{j,k}^c)/\tilde{\Omega}_k^j l_{j,k}^c$ is constant for all particles and depends on the localization function, the number of particles, and $\tilde{\Omega}_k^j$.

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-minimizing SU sampling rather than the merging step, respectively. The procedure for the LWEnKF-SU is as follows:

1. Perform steps 1, 2, 3 of the procedure of the LWEnKF in Section 2.
2. For each scalar observation y_j , perform step 4 of the procedure of the LWEnKF in Section 2, but step (d) is replaced as follows:
 - d. For each model variable k in $\{k | l_{j,k} > 0\}$, perform adjustment-minimizing SU sampling according to the total weights

$$\frac{v_{k,i}^j}{\tilde{\Omega}_k^j}, \quad i = 1, 2, \dots, N.$$

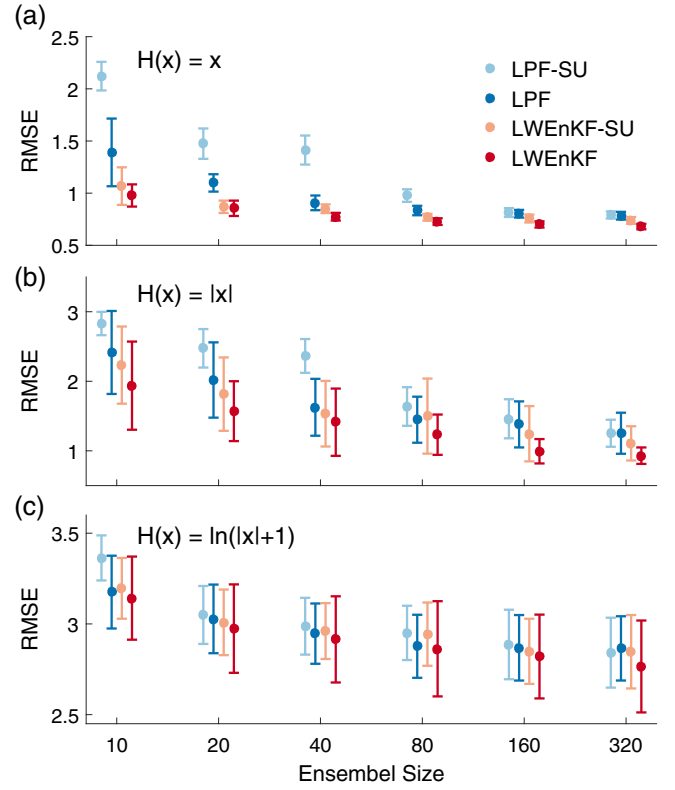


FIGURE 10 The average posterior RMSE over the data assimilation period as functions of ensemble size using the LWEnKF-SU, LPF-SU, LWEnKF and LPF, in which the observation operators are (a) $h(x) = x$, (b) $h(x) = |x|$ and (c) $h(x) = \ln(|x| + 1)$

3. Perform step 5 of the procedure of the LWEnKF in Section 2.

The experimental set-ups in Section 4.1.2 is considered to compare the LWEnKF-SU and LPF-SU with the LWEnKF and LPF. For each observation type, the average RMSEs for each method are shown in Figure A1 as a function of ensemble size. For all types of observations and all particle sizes, the results of the merging method are better than those of the adjustment-minimizing SU sampling method. However, the improvement of the merging step on the LWEnKF is smaller than that on the LPF. When the number of particles is small, the RMSEs of the LPF is significantly lower than that of the LPF-SU. Compared with the nonlinear observation operators, the improvement by the merging step of the LPF is more obvious in the case of the linear observation operator.