We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

6,900
Open access books available

186,000
International authors and editors

200M
Downloads

154
Countries delivered to

Our authors are among the

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

An Introduction to Ensemble-Based Data Assimilation Method in the Earth Sciences

Youmin Tang, Zheqi Shen and Yanqiu Gao

Abstract

In this chapter, the ensemble-based data assimilation methods are introduced, including their developments, applications and existing concerns. These methods include both traditional methods such as Kalman filter and its derivatives and some advanced algorithms such as sigma-point Kalman filters and particle filters. Emphasis is placed on the challenges of applying these methods onto high-dimensional systems in the earth sciences.

Keywords: data assimilation, Kalman filter, EnOI, EnKF, particle filter

1. Introduction

In this chapter, we will talk about the modelling and simulation using both observed data and numerical models, that is, the observations will be incorporated into numerical models for optimal modelling and simulation. In statistics, this is called state-space estimation. In the earth science, it is called data assimilation. For example, a strict definition of data assimilation in atmospheric and oceanic sciences is the process to estimate the state of a dynamic system such as atmospheric and oceanic flow by combining the observational and model forecast data [1].

In general, assimilation methods can be classified into two categories: variational and sequential. This chapter is a tutorial on the sequential data assimilation methods such as ensemble Kalman filter (EnKF) and its variants. A brief introduction of the particle filter (PF) is also provided in this chapter.

This tutorial places emphasis on the rationale behind each method, including: (i) the principle for deriving the algorithm; (ii) the basic assumptions of each method; (iii) the connection and relation between different methods (e.g. extended Kalman filter (EKF) and EnKF, EnKF and sigma-point Kalman filters (SPKF), etc.); and (iv)the advantage and deficiency of each method.

This chapter has been written and organized through teaching for under-/graduatestudents in earth science courses. It can also be a good reference to researchers in the field of modelling and data assimilation.

2. The general framework of several assimilation approaches

Intuitively, one might think that an optimal simulation scheme is to directly replace model variables by observations during numerical integrations. Such a direct replacement is usually not correct since observations are not perfect and contain errors. A simple replacement will introduce observation errors into models, and ignore possible impact of observation errors on model behaviours, easily resulting in imbalance of model dynamics and physics. Thus, the application of observations into numerical models must consider both model and observation errors that play a critical role in the assimilation process.

We will start to display the assimilation concept by a simple example. A detail introduction can be found in [2].

For an unknown true state value, denoted by T_t, there are two samples, denoted by T_1(e.g. model simulation) and T_2(observation), which have the errors ϵ_1 and ϵ_2, respectively. Thus, we have

$$T_1 = T_t + \epsilon_1, \tag{1}$$

$$T_2 = T_t + \epsilon_2. \tag{2}$$

We assume the measurement or observation is unbiased, and the variances of errors are known, i.e. $E(\epsilon_1) = E(\epsilon_2) = 0$, $Var(\epsilon_1) = \sigma_1{}^2$, $Var(\epsilon_2) = \sigma_2{}^2$. The question here is to seek an optimal estimate, denoted by T_a (called analysis in the assimilation field), for T_t using T_1 and T_2. This optimal estimate is the central issue of data assimilation.

There are several methods for this solution, as demonstrated below.

2.1. Least-squares method

Assume the analysis is a linear combination of both T_1 and T_2, that is, $T_a = a_1 T_1 + a_2 T_2$. Due to the assumption that both T_1 and T_2 are unbiased, T_a should be unbiased, i.e. $E(T_a) = E(T_t)$, so $a_1 E(T_1) + a_2 E(T_2) = E(T_t)$, and then $a_1 + a_2 = 1$. The best (optimal) estimate should minimize the variance of T_a as below:

$$\sigma_a^2 = E[T_a - T_t]^2 = E[a_1 T_1 + a_2 T_2 - T_t]^2 = E[a_1(T_1 - T_t) + a_2(T_2 - T_t)]^2$$
$$= E\left(a_1^2 \epsilon_1^2 + a_2^2 \epsilon_2^2 + 2a_1 a_2 \epsilon_1 \epsilon_2\right) = a_1^2 \sigma_1^2 + (1 - a_1)^2 \sigma_2^2, \tag{3}$$

here, we assumed that the errors of T_1 and T_2 are uncorrelated, i.e. $E(\epsilon_1 \epsilon_2)=0$. To minimize σ_a^2, let $\partial \sigma_a^2 / \partial a_1 = 0$, thus

$$a_1 = \frac{\sigma_2^2}{\sigma_1^2 + \sigma_2^2} \tag{4}$$

Namely,

$$T_a = a_1 T_1 + (1 - a_1)T_2 = T_1 + \frac{\sigma_2^2}{\sigma_1^2 + \sigma_2^2}(T_2 - T_1). \tag{5}$$

Using Eq. (5), the variance of T_a could be minimized.

2.2. Variational approach

In general, assimilation methods can be classified into two categories: variational and sequential. Variational methods such as three-dimensional variational (3D-Var) method and four-dimensional variational (4D-Var) method [3, 4] are batch methods, whereas sequential methods such as Kalman filter (KF) [5] belong to the estimation theory.

They both have had great success. The European Centre for Medium-Range Weather Forecasts (ECMWF) introduced the first 4D-Var method into the operational global analysis system in November 1997 [6–8]. The ensemble Kalman filter (EnKF) was first introduced into the operational ensemble prediction system by Canadian Meteorological Centre (CMC) in January 2005 [9].

This chapter is a tutorial of the ensemble-based sequential data assimilation methods, such as EnKF and its variants. However, we will briefly demonstrate the idea of variational assimilation by the above example.

First, a cost function should be defined for variational assimilation approach. For this simple example, we define the cost function as below:

$$J(T) = \frac{1}{2}\left[\frac{(T - T_1)^2}{\sigma_1^2} + \frac{(T - T_2)^2}{\sigma_2^2}\right] \tag{6}$$

$$T = a_1 T_1 + a_2 T_2. \tag{7}$$

The solution is to seek an analysis T_a, determined by a_1 and a_2, leading to the cost function minimum, i.e. $J(T_a) = \min\{J(T)\}$. Obviously, we have $\partial J(T)/\partial a_1 = 0$ and $\partial J(T)/\partial a_2 = 0$. Substitute with (6), it is

$$\frac{\partial J(T)}{\partial a_1} = \frac{T - T_1}{\sigma_1^2}\frac{\partial T}{\partial a_1} + \frac{T - T_2}{\sigma_2^2}\frac{\partial T}{\partial a_1} = 0 \tag{8}$$

Eq. (7) leads to $\dfrac{\partial T}{\partial a_1} = T_1$. Thus, the solution of (8), denoted by T_a, satisfies

$$T_a = \frac{\sigma_2^2}{\sigma_1^2 + \sigma_2^2}T_1 + \frac{\sigma_1^2}{\sigma_1^2 + \sigma_2^2}T_2. \tag{9}$$

The above is a simple example of the 3D variational assimilation approach, where we only consider the analysis error (cost function) for a single time. However, in many cases, we need to consider the error growth during a period, i.e. the sum of errors during the period, in the cost function Eq. (6). For example, the cost function of 4D-Var is defined as below:

$$J(T) = \frac{1}{2}\sum_{n=1}^{N}[\frac{(T(t_n) - T_1(t_n))^2}{\sigma_1^2} + \frac{(T(t_n) - T_2(t_n))^2}{\sigma_2^2}]. \tag{10}$$

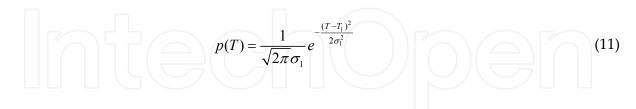Meanwhile $T(t_n)$ follows a dynamical model, saying $T(t_n) = \displaystyle\int_{t_0}^{t_n}F(T(t))dt = M_n(T(t_0))$, where F is a nonlinear dynamical model, M_n is the integral operator and t_0 is the initial time. Thus, the cost function value of (10) is only determined by the initial condition. Namely, the objective here is to seek optimal initial condition $T(t_0)$ that enables (10) minimum, i.e. minimizing (10) subject to dynamical model F. This is a standard conditional extreme problem that can be solved by Lagrange multiplier approach. However, the complexity of dynamical model excludes the possibility to get the analytical solution. We have to solve the minimum problem with aid of numerical methods, e.g. Newton conjugate gradient method. All of numerical methods require the gradient value $\dfrac{\partial J}{\partial T_0}$ for solution.

Again, it is almost impossible for obtaining analytical solution of $\dfrac{\partial J}{\partial T_0}$ due to the complexity of F. Usually researchers get the gradient value numerically using an approach of tangent linear and adjoint models. The details on tangent linear and adjoint models can be found in relevant references as cited above. It should be noticed that it is very difficult, even intractable some-times, to construct tangent linear and adjoint models in some cases. Thus, more and more researchers have started to apply sequential assimilation methods instead of 4D-Var in recent years. Next, we will introduce the concept of the sequential assimilation method using the above example.

2.3. Bayesian approach

Assume T_1 and σ_1 are the mean value and standard deviation of the model prediction that implies a prior probability distribution of truth T,

$$p(T) = \frac{1}{\sqrt{2\pi}\,\sigma_1} e^{-\frac{(T-T_1)^2}{2\sigma_1^2}} \tag{11}$$

Obviously, this is a Gaussian distribution function, which can be denoted by $N(T_1, \sigma_1)$ Given the observation T_2 and its standard deviation σ_2, the posterior distribution of the truth can be expressed by Bayes' theorem:

$$p(T|T_2) = \frac{p(T_2|T)\,p(T)}{p(T_2)} \propto \frac{1}{\sqrt{2\pi}\,\sigma_2} e^{\frac{(T-T_2)^2}{2\sigma_2^2}} \frac{1}{\sqrt{2\pi}\,\sigma_1} e^{\frac{(T-T_1)^2}{2\sigma_1^2}}. \tag{12}$$

$p(T_2)$ was ignored in (12) since it is independent of T, and usually plays as a normalization factor. The likelihood function $p(T_2|T)$ describes the probability that the observation becomes T_2 given an estimation of T. It is commonly assumed to be Gaussian $N(T, \sigma_2)$. The object here is to estimate the truth by maximizing the posterior probability $p(T|T_2)$(namely, we ask the truth to occur as much as possible—maximum probability). Maximizing $p(T|T_2)$ is equivalent to maximizing the logarithm of the right item of (12), i.e.

$$\begin{aligned}
\log(p(T|T_2)) &= \log(\frac{1}{\sqrt{2\pi}\,\sigma_2}) - \frac{(T-T_2)^2}{2\sigma_2^2} + \log(\frac{1}{\sqrt{2\pi}\,\sigma_1}) - \frac{(T-T_1)^2}{2\sigma_1^2} \\
&= \text{const} - \frac{1}{2}\left[\frac{(T-T_2)^2}{\sigma_2^2} + \frac{(T-T_1)^2}{\sigma_1^2}\right].
\end{aligned} \tag{13}$$

Obviously, the maximum of $p(T|T_2)$ occurs at the minimum of the second item on the right-hand side of (13), i.e. the minimum of the cost function J of (6). Thus, under the assumption of Gaussian distribution, maximizing a posterior probability (Bayesian approach) is equivalent to minimizing cost function (variational assimilation approach). Further, if the model F is linear and the probability distribution is Gaussian, it can be further proved that the Kalman filter is equivalent to 4D-Var adjoint assimilation method.

3. Optimal interpolation (OI) and Kalman filter (KF)

3.1. Optimal interpolation

The most special case in data assimilation is that the forecast model is linear and the errors are Gaussian. The solution among sequential methods to this case is provided by Kalman filter. Typically, the Kalman filter applies to the below state-space model:

$$x_{t+1} = Mx_t + \eta_t, \tag{14}$$

$$y_t = Hx_t + \zeta_t, \tag{15}$$

where M and H are linear operators of model and measurement, respectively. x is model state and y is the observation, and the subscript implies the time step. η_t and ζ_t are the model errors and observational errors, respectively, which have variance: $var(\eta_t) = <\eta_t, \eta_t^T> = Q, var(\eta_t) = <\zeta_t, \zeta_t^T> = R$. The objective here is to estimate model state x using y, making it close to true state (unknown) as much as possible.

Assuming the estimate of model state x^a at a time step is a linear combination of model forecast x^b and observation y^o, i.e. the filter itself is linear, so

$$x^a = x^b + K\left[y^o - Hx^b\right]. \tag{16}$$

Eq. (16) is the standard expression of Kalman filter. K is called Kalman gain that determines the optimal estimate and $y^o - Hx^b$ is called the innovation. An analysis step is essentially to determine the increment to the forecast by combining the Kalman gain and the innovation. Before deriving K, we denote the covariance matrix of the analysis error ϵ^a by P^a, i.e. $P^a = <\epsilon^a, (\epsilon^a)^T>$, where $\epsilon^a = x^a - x^{tr}$ and x^{tr} is the true value of model state. Similarly, observed errors and forecast errors are defined by $\epsilon^o = y^o - Hx^{tr}$ and $\epsilon^b = x^b - x^{tr}$, respectively. It should be noticed that the forecast error ϵ^b is different from the model error ζ_t that is a systematic bias. Also, we denote $B = <\epsilon^b, (\epsilon^b)^T>$ as the background (forecast) error covariance and $R = <\epsilon^o, (\epsilon^o)^T>$ as the observational error covariance. It is also assumed that the observation error is not related to forecast error, so $<\epsilon^b, (\epsilon^o)^T> = <\epsilon^o, (\epsilon^b)^T> = 0$.

Clearly, we are seeking for K that can lead to P^a minimum. Subtracting x^{tr} on both sides of Eq. (16) leads to the below equation:

$$x^a - x^{tr} = x^b - x^{tr} + K\left[y^o - Hx^b + Hx^{tr} - Hx^{tr}\right]. \tag{17}$$

Namely,

$$\epsilon^a = \epsilon^b + K\left(\epsilon^o - H\epsilon^b\right), \tag{18}$$

And

$$
\begin{aligned}
P^a &= E\left[\epsilon^b + K\left(\epsilon^o - H\epsilon^b\right)\right]\left[\epsilon^b + K\left(\epsilon^o - H\epsilon^b\right)\right]^T \\
&= E\left[\epsilon^b\left(\epsilon^b\right)^T + \epsilon^b\left(\epsilon^o - H\epsilon^b\right)^T K^T + K\left(\epsilon^o - H\epsilon^b\right)\left(\epsilon^b\right)^T + K\left(\epsilon^o - H\epsilon^b\right)\left(\epsilon^o - H\epsilon^b\right)^T K^T\right] \\
&= B - BH^T K^T - KHB + K\left(R + HBH^T\right)K^T.
\end{aligned}
\tag{19}
$$

Here, we used $B = B^T$. The optimal estimate asks the trace of P^a minimum, namely, $\partial\left[trace\left(P^a\right)\right]/\partial K = 0$. It can be computed that

$$K = BH^T (HBH^T + R)^{-1}. \tag{20}$$

Substitute into (13)

$$P^a = B - BH^T K^T - KHB + BH^T (HBH^T + R)^{(-1)}(R + HBH^T)K^T = (I - KH)B. \tag{21}$$

Here, we invoked the below properties:

$$\frac{\partial Ax}{\partial x^T} = \frac{\partial x^T A}{\partial x} = A \tag{22}$$

$$\frac{\partial x^T Ax}{\partial x} = \left(A + A^T\right)x \tag{23}$$

$$\frac{\partial A^T x}{\partial x^T} = \frac{\partial x^T A^T}{\partial x} = A^T \tag{24}$$

$$\frac{\partial(\text{trace}[XAX^T])}{\partial x} = X(A + A^T) \tag{25}$$

$$\frac{\partial(\text{trace}[AX^T])}{\partial x} = A^T \qquad (26)$$

Thus, we have the optimal estimate filter:

$$x^a = x^b + K\left[y^o - Hx^b\right], \qquad (27)$$

$$K = BH^T(HBH^T + R)^{-1}, \qquad (28)$$

$$P^a = (I - KH)B. \qquad (29)$$

In the estimate (27)–(29), if the background error covariance B is prescribed, the estimate is called optimal interpolation. The OI does not involve state equation (14) and B is unchanged during the entire assimilation process.

3.2. Kalman filter

Now, we consider that B in (28) changes with the assimilation cycle. This is more realistic since the model prediction errors should be expected to decrease with the assimilation.

From Eq. (14), we have

$$x_{t+1}^{tr} = Mx_t^{tr} + \eta_t, \qquad (30)$$

$$x_{t+1}^b = E(Mx_t^a + \eta_t) = Mx_t^a \qquad (31)$$

Eq. (30) indicates that even the true value is input at a time step, model cannot get a true value for next step due to model bias η_t. Eq. (31) shows a standard procedure for the model prediction of next step starting from the analysis of previous step.

Subtracting (30) from (31) produces

$$\epsilon_{t+1}^b = M\epsilon_t^a - \eta_t, \qquad (32)$$

$$B_{t+1} = E\left(\epsilon_{t+1}^b\left(\epsilon_{t+1}^b\right)^T\right) = E[(M\epsilon_t^a + \eta_t)(M\epsilon_t^a + \eta_t)^T] = MP_t^aM^T + Q \qquad (33)$$

where $P_t^a = < \epsilon_t^a , (\epsilon_t^a)^T >$ represents the analysis error covariance for time step t. The above equation considers the evolution of the background (prediction) error covariance with the time controlled by the dynamical model operator M. The above equations constitute the framework of Kalman filter (**Table 1**), namely

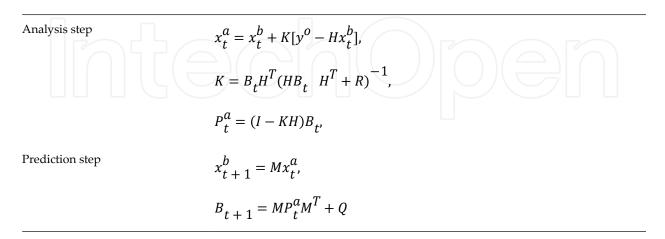| Analysis step	$x_t^a = x_t^b + K[y^o - Hx_t^b],$
	$K = B_t H^T (HB_t \ H^T + R)^{-1},$
	$P_t^a = (I - KH)B_t,$
Prediction step	$x_{t+1}^b = Mx_t^a,$
	$B_{t+1} = MP_t^a M^T + Q$

Table 1. The Kalman filter.

One Kalman filter cycle consists of two parts, namely, one analysis step (Eqs. (27)–(29)) and one prediction step (Eqs. (31) and (33)). The analysis state x_t^a and covariance P_t^a are treated as initial conditions for the prediction step, until the next observation is available. Sometimes, B_t is denoted by P_t^f in Kalman filter literatures.

3.3. Extended Kalman filter (EKF)

In deriving the Kalman filter, we assume the state model M and measurement model H are both linear. Further, we also assume the error has Gaussian distribution. Therefore, classic KF only works for linear models and Gaussian distribution. If the dynamical model and measurement model are not linear, we cannot directly apply KF. Instead, linearization must be performed prior to apply KF. The linearized version of KF is called extended KF (EKF), which solves the below state-space estimate problem:

$$x_{t+1} = f(x_t) + \eta_t, \tag{34}$$

$$y_t = h(x_t) + \zeta_t, \tag{35}$$

where f and h are nonlinear models, and η_t and ζ_t are additive noises.

The filter is still assumed to be 'linear', i.e.

$$x^a = x^b + K[y^o - h(x^b)] \tag{36}$$

Actually, it is not a linear combination of the forecast x^b and observation y^o if his not linear. However, we just extend the formulation of Eq. (16), and apply it intuitively in nonlinear cases. Ignoring high-order terms, the following holds approximately

$$h(x + \delta x) = h(x) + \frac{\partial h}{\partial x} \delta x = h(x) + H\delta x \tag{37}$$

where H is the linearization of h and $H_{i,j} = \dfrac{\partial h_i}{\partial x_j}$. So,

$$y^o - h(x^b) = y^o - h(x^{tr} + x^b - x^{tr}) = y^o - h(x^{tr}) - H(x^b - x^{tr}) = \epsilon^o - H\epsilon^b \tag{38}$$

$$x^a = x^b + K(\epsilon^o - H\epsilon^b) \tag{39}$$

Eq. (39) is identical to Eq. (16). Similarly, subtracting x^{tr} on both sides of Eq. (47) leads to the below equation:

$$\epsilon^a = \epsilon^b + K(\epsilon^o - H\epsilon^b) \tag{40}$$

which is the same as Eq. (18). Following the same derivation as that for Eq. (18), we can obtain the equations similar to (27)–(29). Therefore, if the measurement model h is nonlinear, the KF can be still applied with a linearization of h.

Similar to Eqs. (30) and (31), the state model is as below:

$$x_{t+1}^{tr} = f(x_t^{tr}) + \eta_t \tag{41}$$

$$x_{t+1}^f = E(f(x_t^a) + \eta_t) = f(x_t^a). \tag{42}$$

Subtracting Eq. (41) from Eq. (42) produces

$$
\begin{aligned}
\epsilon_{t+1}^f &= f(x_t^a) - f(x_t^{tr}) - \eta_t = f(x_t^a) - f(x_t^{tr} - x_t^a + x_t^a) - \eta_t \\
&= f(x_t^a) - f(x_t^a - \epsilon_t^a) - \eta_t = M\epsilon_t^a - \eta_t
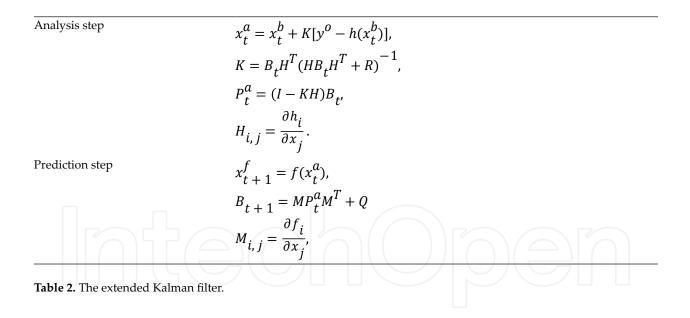\end{aligned}
\tag{43}
$$

where $M_{i,j} = \dfrac{\partial f_i}{\partial x_j}$.

Comparing Eq. (31) with Eq. (33), it reveals that Eq. (33) still works. Thus, the EKF can be summarized as below (**Table 2**).

The procedure to perform EKF is similar to that for KF, as listed above. The disparities and similarities between EKF and KF include

i.   Kalman gain K has the same form for both, especially the linear or linearized measurement model should be used;

ii.   the update equation of model error covariance has the same form, with linear and linearized state model used;

iii. forecast model is different, with KF using linear Eq. (14) and EKF using nonlinear model Eq. (34); and

iv.   the filtering algorithm is different, linear measurement model H used in KF and nonlinear model h in EKF.

It should be noticed that EKF is only an approximate of KF for nonlinear state model.

Analysis step	$x_t^a = x_t^b + K[y^o - h(x_t^b)],$
	$K = B_t H^T (H B_t H^T + R)^{-1},$
	$P_t^a = (I - KH) B_t,$
	$H_{i,j} = \dfrac{\partial h_i}{\partial x_j}.$
Prediction step	$x_{t+1}^f = f(x_t^a),$
	$B_{t+1} = M P_t^a M^T + Q$
	$M_{i,j} = \dfrac{\partial f_i}{\partial x_j},$

Table 2. The extended Kalman filter.

4. Ensemble Kalman filter (EnKF)

4.1. Basics of EnKF

A challenge in EKF is to update background (prediction) error covariance, which requires the linearization of nonlinear model. The linearization of nonlinear model is often difficult technically, and even intractable in some cases, e.g. non-continuous functions existing in

models. Another drawback of EKF is to neglect the contributions from higher-order statistical moments in calculating the error covariance.

To avoid the linearization of nonlinear model, the ensemble Kalman filter (EnKF) was introduced by Evensen [10, 11], in which the prediction error covariance B of Eq. (33) are estimated approximately using an ensemble of model forecasts. The main concept behind the formulation of the EnKF is that if the dynamical model is expressed as a stochastic differential equation, the prediction error statistics, which are described by the Fokker-Flank equation, can be estimated using ensemble integrations ([10, 12]; thus, the error covariance matrix B can be calculated by integrating the ensemble of model states. The EnKF can overcome the EKF drawback that neglects the contributions from higher-order statistical moments in calculating the error covariance. The major strengths of the EnKF include the following:

i.   there is no need to calculate the tangent linear model or Jacobian of nonlinear models, which is extremely difficult for ocean (or atmosphere) general circulation models (GCMs);

ii.   the covariance matrix is propagated in time via fully nonlinear model equations (no linear approximation as in the EKF); and

iii.   it is well suited to modern parallel computers (cluster computing) [13].

EnKF has attracted a broad attention and been widely used in atmospheric and oceanic data assimilation.

Simply saying, EnKF avoids the computation and evolution of the error covariance B as in Eq. (33), and computes B using below formula as soon as it is required.

$$B = \frac{1}{N-1}\sum_{i=1}^{N}(x_i^b - \overline{x^b})(x_i^b - \overline{x^b})^T \tag{44}$$

where x_i^b represents the i-th member of the forecast ensemble of system state vector at step t, and N is the ensemble size. The use of Eq. (44) avoids processing M, the linearized operator of nonlinear model. However, the measurement function H is still linear or linearized while computing the Kalman gain K, which causes concern. To avoid the linearization of nonlinear measurement function, Houtekamer and Mitchell [14] wrote Kalman gain by

$$K = BH^T(HBH^T + R)^{-1}, \tag{45}$$

$$BH^T \equiv \frac{1}{N-1}\sum_{i=1}^{N}[x_i^b - \overline{x^b}][h(x_i^b) - \overline{h(x^b)}]^T, \tag{46}$$

$$HBH^T \equiv \frac{1}{N-1}\sum_{i=1}^{N}[h(x_i^b) - \overline{h(x^b)}][h(x_i^b) - \overline{h(x^b)}]^T, \tag{47}$$

where $\overline{h(x^b)} = \frac{1}{N}\sum_{i=1}^{N} h(x_i^b)$. Eqs. (46) and (47) allow direct evaluation of the nonlinear measurement function h in calculating Kalman gain. However, Eqs. (46) and (47) have not been proven mathematically, and only were given intuitively. Tang and Ambadan argued that Eqs. (46) and (47) approximately hold if and only if $\overline{h(x^b)} = h(\overline{x^b})$ and $x_i^b - \overline{x^b}$ is small for $i = 1, 2, ..., N$ [15]. Under these conditions, Tang et al. argued Eqs. (46) and (47) actually linearize the nonlinear measurement functions h to H [16]. Therefore, direct application of the nonlinear measurement function in Eqs. (46) and (47), in fact, imposes an implicit linearization process using ensemble members. In next section, we will see that Eqs. (46) and (47) can be modified under a rigorous framework.

Thus, the procedures of EnKF are summarized as below (**Table 3**):

1. Imposing perturbations on initial conditions and integrate the model, i.e. $x_{i, 1} = f(x_0 + \gamma_i)$, where $i = 1, 2..., N$(ensemble size) and x_0 is the initial condition.

2. Using $K = BH^T(HBH^T + R)^{-1}$ and Eqs. (46) and (47) to calculate Kalman gain K.

3. Calculating analysis using

$$x_i^a = x_i^b + K\left[y^o + \varepsilon^i - h\left(x_i^b \right) \right], \tag{48}$$

   after K is obtained. It should be noted that each ensemble member produces an analysis; the average of all (N) analyses can be obtained.

4. Using $x_{i, t+1}^b = f(x_i^a)$ to obtain new ensemble members for next round analysis.

5. Repeating Steps 2–4 until the end of assimilation period.

Analysis step	$x_i^a = x_i^b + K[y^o + \varepsilon^i - h(x_i^b)], i = 1, ..., N$
	$K = BH^T(HBH^T + R)^{-1}$,
	$BH^T = \frac{1}{N-1}\sum_{i=1}^{N} [x_i^b - \overline{x^b}][h(x_i^b) - \overline{h(x^b)}]^T$,
	$HBH^T = \frac{1}{N-1}\sum_{i=1}^{N} [h(x_i^b) - \overline{h(x^b)}][h(x_i^b) - \overline{h(x^b)}]^T$
Prediction step	$x_{i, t+1}^b = f(x_i^a + \gamma_i), \quad i = 1, ..., N$

Table 3. The ensemble Kalman filter.

It should be noted that the observation should be treated as a random variable with the mean equal to y^o and covariance equal to R. This is why there is ε_i in Eq. (48). Simply, ε_i is often drawn from a normal distribution $\varepsilon_i \sim N(0, R)$.

From the above procedure, we find that Eq. (44) is not directly applied here. Instead, we use Eqs. (46) and (47) to calculate K. This is because Eqs. (46) and (47) avoid the linearization of nonlinear model and also avoid the explicit expression of matrix B, which is often very large and cannot be written in current computer sources in many realistic problems. The measurement function, h, projecting model space (dimension) to observation space (dimension), greatly reduces the number of dimension.

4.2. Some remarks on EnKF with large dimensional problems

4.2.1. Initial perturbation

The success of EnKF highly depends on the quality of ensemble members produced by initial perturbations. It is impractical to represent all possible types of errors within the ensemble because of the computational cost, the method of generating initial perturbations must be chosen judiciously.

The first issue is the amplitude of initial perturbations. Usually, the following two factors are considered when selecting the amplitude of initial perturbations: the amplitude of observation error and the amplitude of model errors induced by model parameters and uncertainty in model physics. If a model is perfect, the amplitude of the perturbations should be the same as the amplitude of observation errors. This combined error is used to determine the amplitude of perturbations.

When the perturbation amplitude is determined, the practical initial perturbation field generating each ensemble member could be constructed by a normalized pseudorandom field multiplied by the prescribed amplitude. Considering the spatial coherence, the pseudorandom field is red noise as proposed by Evensen [17], summarized as below:
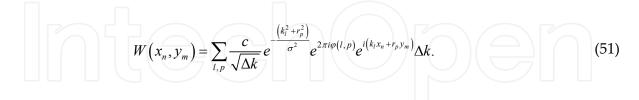
1. Calculate the statistical characteristics for the pseudorandom field to meet its variance of 1 and mean of 0 by solving the following nonlinear equation:

$$e^{-1} = \frac{\sum_{l,p} e^{-2(k_l^2 + r_p^2)/\sigma^2} \cos(k_l r_h)}{\sum_{l,p} e^{-2(k_l^2 + r_p^2)/\sigma^2}}, \tag{49}$$

where $k_l = \frac{2\pi l}{x_n} = \frac{2\pi l}{N_x \Delta x}$, $r_p = \frac{2\pi p}{y_m} = \frac{2\pi p}{N_y \Delta y}$, and N_x and N_y are the number of grid points in the x-axis (lon.) and the y-axis (lat.). The l and p are wavenumbers, changing from 1 to the maximum value of $N/2$ and $M/2$. Δx and Δy are the intervals of two adjacent points, often set to 1, and r_h is the decorrelation length. The purpose of Eq. (49) is to derive σ^2 for the other feature:

$$c^2 = \frac{1}{\Delta k \sum_{l,p} e^{-2(k_l^2 + r_p^2)/\sigma^2}}$$ (50)

2. After c and σ^2 are obtained, we can construct a two-dimensional pseudorandom field:

$$W(x_n, y_m) = \sum_{l,p} \frac{c}{\sqrt{\Delta k}} e^{-\frac{(k_l^2 + r_p^2)}{\sigma^2}} e^{2\pi i \varphi(l,p)} e^{i(k_l x_n + r_p y_m)} \Delta k.$$ (51)

3. While x_n, y_m cover the whole domain, Eq. (51) produces a $N_x * N_y$ two-dimensional random field with spatial coherence structure and the variance of 1 and mean of 0. If the realistic uncertainty (error) has an amplitude β, the perturbation should be βW. Similarly, Eq. (51) is often used for the error perturbation γ_i used in the fourth step of the EnKF procedure.

Sometimes, we need to consider the vertical coherence of pseudorandom fields between adjacent levels in oceanic models. A simple method for this purpose is to construct the pseudorandom field at the kth level ε_k by following equation:

$$\varepsilon_k = \alpha \varepsilon_{k-1} + \sqrt{1 - \alpha^2} W_k,$$ (52)

where $W_k (k = 1, ..., N_z)$ is the pseudorandom field at the kth level without considering vertical coherence, constructed using the above method. Initially, for the surface perturbation ($k = 1$), the vertical coherence is not considered, i.e. equals to zero since ε_{k-1} does not exist. Eq. (52) indicates that a practical pseudorandom at the kth level (ε_k) is composed of W_k and ε_{k-1}. As such the ε_k is correlated with ε_{k-1}, i.e. the practical pseudorandom fields of two adjacent levels (ε_{k-1} and ε_k) are coherent with each other. Their correlation or coherent structure is determined by the coefficient $\alpha \in [0, 1]$. Eq. (52) generates a sequence that is white in the vertical direction if $\alpha = 0$ (i.e. $\varepsilon_k = W_k$), but a sequence that is perfect correlated in vertical if $\alpha = 1$ (i.e. $\varepsilon_k = \varepsilon_{k-1}$). Eq. (52) is also often used to construct random field that is temporally coherent, for example, a continuous random noise that has coherence in time, as used for γ_i in the forecast model [17]. The random noise γ_i in the EnKF procedure can also be replaced by the random noise imposed in model forcing. For example, the random noise is continuously added to wind forcing for oceanic models. Even for some atmospheric models with transition processes, there are inherent random noises making γ_i not necessary. One important criteria for γ_i and the amplitude β is to examine ensemble spread by some sensitivity experiments.

4.2.2. The computational cost of Kalman gain

The Kalman gain K has dimension of $L * m$, where L is the number of model variables and m is the number of observational variables. In many realistic problems, L and m are very large numbers ($m \gg N$, the ensemble size), making the inversion very expensive.

A simple procedure is to rewrite the Kalman gain K, as below:

$$K = \tilde{x}\tilde{x}^T H^T (H\tilde{x}\tilde{x}^T H^T + \varepsilon\varepsilon^T)^{-1}, \tag{53}$$

where $\tilde{x}$ indicates that the model ensemble predictions removed the ensemble mean ($\tilde{x}_i = [x_i^b - \overline{x^b}]$, for $i = 1, 2, ..., N$). $R = \frac{1}{N}\varepsilon\varepsilon^T$ was invoked here. If we assume the ensemble prediction error ($x^b - \overline{x^{tr}} \approx x^b - \overline{x^b} = \tilde{x}$) is not correlated to observation error, i.e. $\tilde{x}\varepsilon^T = 0$, the following is valid [17]:
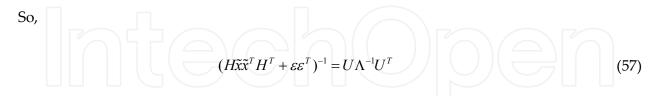
$$(H\tilde{x}\tilde{x}^T H^T + \varepsilon\varepsilon^T) = (H\tilde{x} + \varepsilon)(H\tilde{x} + \varepsilon)^T, \tag{54}$$

where $(H\tilde{x} + \varepsilon)$ has dimension $m * N$. Usually, ensemble size N is much less than m. Using the singular-value decomposition (SVD) technique, we have

$$(H\tilde{x} + \varepsilon) = U\Sigma V^T \tag{55}$$

Eq. (54) then becomes

$$(H\tilde{x}\tilde{x}^T H^T + \varepsilon\varepsilon^T) = U\Sigma V^T V\Sigma^T U = U\Sigma\Sigma^T U^T = U\Lambda U^T \tag{56}$$

So,

$$(H\tilde{x}\tilde{x}^T H^T + \varepsilon\varepsilon^T)^{-1} = U\Lambda^{-1}U^T \tag{57}$$

where U and Λ are the eigenvector and the square of eigenvalues of $(H\tilde{x} + \varepsilon)$. There are N non-zero eigenvalues for $(H\tilde{x} + \varepsilon)$, therefore the dimension is not large, allowing us to efficiently compute the inversion for a global analysis in most practical situations.

4.2.3. Stochastic EnKF and deterministic EnKF

In EnKF introduced in the previous section, the observation assimilated into dynamical model should be treated to be stochastic variable, as expressed by $y^o + \varepsilon^i$ in Eq. (48). It is a must if the classic EnKF algorithm is used. It has been proven that if the EnKF assimilates deterministic

observations (i.e., observation y^o not changed at each ensemble member), the analysis error covariance will be systematically underestimated, typically leading to filter divergence, as indicated by below [11, 18]:

$$P^{a*} = (I - KH)B(I - KH)^T \tag{58}$$

Eq. (58) gives the analysis error covariance if the observed is not perturbed. Comparing Eq. (58) with Eq. (29), a theoretically unbiased estimate, P^{a*} is always less than P^a.

However, the perturbed observation approach (i.e. $y^o + \varepsilon^i$) introduces an additional source of sampling error that reduces analysis error covariance accuracy and increases the probability of understanding analysis error covariance [19, 20]. Thus, an approach that only uses a single observation realization but avoids systematical underestimation of analysis error covariance was pursued. There are several approaches to implement this goal, as summarized by Tippett et al. [20]. Below, we will introduce an approach developed by Whitaker and Hamill [19], called Ensemble squareroot filter (EnSRF).

Denote the deviation of analysis from the analysis mean by $\tilde{x}^a = x^a - \bar{x}^a$, it is easy to write

$$\tilde{x}^a = \tilde{x}^b + \tilde{K}\left[\tilde{y}^o - H\tilde{x}^b \right] \tag{59}$$

where $\tilde{y}^o = y^o - \bar{y}^o$. If a single observation realization is assimilated in all ensemble members, $\tilde{y}^o = 0$ and

$$\tilde{x}^a = \tilde{x}^b - \tilde{K}H\tilde{x}^b = \left(I - \tilde{K}H \right)\tilde{x}^b, \tag{60}$$

$$P^{a*} = \left(I - \tilde{K}H \right)B\left(I - \tilde{K}H \right)^T. \tag{61}$$

We seek a definition for $\tilde{K}$ that will result in an ensemble whose analysis error covariance equals to $(I - KH)B$, i.e.

$$(I - \tilde{K}H)B(I - \tilde{K}H)^T = \left(I - KH \right)B. \tag{62}$$

The solution of Eq. (62) is

$$\tilde{K} = (1 + \sqrt{\frac{R}{HBH^T + R}})^{-1} K. \tag{63}$$

Therefore, EnSRF is summarized as below (**Table 4**):

$$\overline{x}^a = \overline{x^b} + K\left[y^o - H\overline{x^b}\right]$$

$$\tilde{x}^a = \tilde{x}^b - \tilde{K}H\tilde{x}^b$$

$$x^a = \overline{x}^a + \tilde{x}^a$$

$$K = BH^T(HBH^T + R)^{-1},,$$

$$\left[\begin{array}{c} \\ BH^T \\ \\ \end{array}\right] = \frac{1}{N-1}\sum_{i=1}^{N}[x_i^b - \overline{x^b}][h(x_i^b) - \overline{h(x^b)}]^T$$

$$HBH^T = \frac{1}{N-1}\sum_{i=1}^{N}[h(x_i^b) - \overline{h(x^b)}][h(x_i^b) - \overline{h(x^b)}]^T$$

$$\tilde{K} = (1 + \sqrt{\frac{R}{HBH^T + R}})^{-1} K$$

Table 4. The analysis scheme of EnSRF.

It should be noted that there are two Kalman gains used in EnSRF, the original K for updating ensemble mean and a new $\tilde{K}$ for updating the anomalies. It indicates that one single observation realization of classic EnKF has the same ensemble analysis mean as stochastic observations.

Initially, the term EnKF refers, in particular, to the stochastic ensemble Kalman filter that requires perturbing the observations. Subsequently, several deterministic EnKFs that avoid the use of perturbed observations were developed, e.g. the ETKF [21], the EAKF [22] and the EnSRF. These filter designs are labelled as variants of the EnKF because they are also based on the Kalman filtering formula and ensemble representations.

4.2.4. Inflation approach

The forecast error covariance is defined by (44)

$$B = \frac{1}{N-1}\sum\nolimits_{i=1}^{N}(x_i^b - \overline{x^b})(x_i^b - \overline{x^b})^T = \frac{1}{N-1}\tilde{X} * \tilde{X}^T.$$ (64)

Eq. (64) is an approximation to B using forecast ensemble. Due to limited computational source, the ensemble size N is often restricted to a small value for many realistic issues. A small ensemble size may cause a very small ensemble spread, causing the approximation of B by Eq. (64), which is seriously underestimated. To solve this problem, B is multiplied by an inflator factor λ (slightly greater than 1). λ is empirically determined, such as some sensitivity experiments, with the typical value of 1.01. λB is used to replace B in EnKF formula. This approach is equivalent to the below approach:

$$x_i^b = \lambda\left(x_i^b - \overline{x^b}\right) + \overline{x^b}$$ (65)

4.2.5. Localization of EnKF

When EnKF is applied to high-dimensional atmospheric and oceanic models, the limited ensemble size will cause the estimated correlations to be noisy [11]. When the ensemble size is insufficient, it will produce spurious correlations between distant locations in the background covariance matrix B. Unless they are suppressed, these spurious correlations will cause observations from one location to affect the analysis in locations an arbitrarily large distance away, in an essentially random manner [23]. This needs to be remedied by the localization method.

Another reason for using localization is that the treatment of localization artificially reduces the spatial domain of influence of observations during the update. The localization dramatically reduces the necessary ensemble size, which is very important for operational systems. Two most common distance-based localization methods used in practice are local analysis and covariance localization.

Using local analysis, only measurements located within a certain distance from a grid point will impact the analysis in this grid point. This allows for an algorithm where the analysis is computed grid point by grid point. It was found that severe localization could lead to imbalance, but with large enough radius of influence (decorrelation length) for the measurements, this was not a problem. Hunt et al. use the local analysis method in their ETKF algorithm and developed a local ensemble transform Kalman filter (LETKF) [23].

To eliminate the small background error covariance associated with remote observations, Houtekamer and Mitchell uses a Schur (element-wise) product of a correlation function with local support and the covariance of the background error calculated from the ensemble [14]. That is, the matrix B in Eq. (48) is replaced by $\rho \circ B$, where "∘" represents the element-wise

product and the elements ρ relates to the distance r of the grid point to the observation r as below:

$$\rho(r) = \left(1 + \alpha r + \frac{\alpha^2 r^2}{3}\right) e^{-\alpha r}. \tag{66}$$

Here, α is a scalar parameter. To the best of author's knowledge, this is the first case that the covariance localization is used in EnKF.

Nowadays, a typical covariance localization approach is used to represent prior covariances using an element-wise product of ensemble covariance and a correlation function with compact support [24]. Anderson applied this approach to the Data Assimilation Research Testbed system [25], which has been used for realistic cases.

5. General form of ensemble-based filters for Gaussian models

In proceeding sections, we introduced Kalman-based filters. Originally Kalman filter applies linear model and linear measurement function. Further, EKF and EnKF were developed to address nonlinear models. However, the measurement functions are still assumed to be linear. Eqs. (46) and (47) can directly evaluate nonlinear measurement functions but they were proposed intuitionally and not proven yet. In this section, we will present a general form for nonlinear measurement function and further prove Eqs. (46) and (47) mathematically using the general form.

For generality, we assume the nonlinear model as Eqs. (34) and (35):

$$x_{t+1} = f\left(x_t\right) + \eta_t, \tag{67}$$

$$y_t = h\left(x_t\right) + \zeta_t, \tag{68}$$

where f and h are nonlinear operators of model and measurement. x is model state and y is the observation. η_t and ζ_t are the model errors and observed errors, respectively, which have variance $var(\eta_t) = <\eta_t, \eta_t^T> = Q, var(\eta_t) = <\zeta_t, \zeta_t^T> = R$. Assuming the estimate of model state x^a at a time step is a linear combination of model forecast x^b and observation y^o, i.e. the filter itself is linear, so

$$x^a = x^b + K\left[y^o - h\left(x^b\right)\right] \tag{69}$$

Denoting $\hat{x}^a = x^t - x^a$, $\hat{x}^b = x^t - x^b$, $\hat{y} = y^o - h(x^b)$, we have

$$\hat{x}^a = \hat{x}^b - K\hat{y} \tag{70}$$

$$
\begin{aligned}
P^a = E[\hat{x}^a(\hat{x}^a)^T] &= E\left[\left(\hat{x}^b - K\hat{y}\right)\left(\hat{x}^b - K\hat{y}\right)^T\right] \\
&= E[\hat{x}^b(\hat{x}^b)^T - \hat{x}^b\hat{y}^T K^T - K\hat{y}(\hat{x}^b)^T + K\hat{y}\hat{y}^T K^T] \\
&= P^b - P_{\hat{x}\hat{y}}K^T - KP_{\hat{y}\hat{x}} + KP_{\hat{y}\hat{y}}K^T
\end{aligned}
\tag{71}
$$

The optimal estimate asks the trace of P^a minimum, namely,

$$\frac{\partial[\text{trace}(P^a)]}{\partial K} = -P_{\hat{x}\hat{y}} - P_{\hat{y}\hat{x}} + 2KP_{\hat{y}\hat{y}} = 0, \tag{72}$$

where we invoked the below properties:

$$\frac{\partial\left(\text{trace}\left[XAX^T\right]\right)}{\partial X} = X\left(A + A^T\right) = 2XA, \tag{73}$$

$$\frac{\partial(\text{trace}[XA^T])}{\partial X} = \frac{\partial(\text{trace}[AX^T])}{\partial X} = A^T = A., \tag{74}$$

Thus, we have the optimal estimate filter:

$$x_t^a = x_t^b + K\left[y^o - h\left(x_t^b\right)\right] \tag{75}$$

$$K = P_{\hat{x}\hat{y}}P_{\hat{y}\hat{y}}^{-1} \tag{76}$$

$$P^a = P^b - KP_{\hat{x}\hat{y}}, \tag{77}$$

Eqs. (75)–(77) give a general algorithm for Gaussian nonlinear model and nonlinear measurement function. The first term of Eq. (74) can be interpreted as the cross-covariance $P_{\hat{x}\hat{y}}$ between the state and observation errors, and the remaining expression can be interpreted as the

error covariance $P_{\hat{y}\hat{y}}$ of the difference between model observation and observation itself. Here, $\hat{y}$ is defined as the error between the noisy observation y^o and its prediction $h(x^b)$.

If the model is linear, obviously,

$$x_{t+1}^b = Mx_t^a + \eta_t,, \tag{78}$$

$$B_{t+1} = MP_t^a M^T + Q. \tag{79}$$

If the measurement function is linear, i.e.

$$\hat{y} = y^o - h(x^b) - \zeta = y^o - Hx^b - \zeta = Hx^{tr} - Hx^b - \zeta = H\hat{x}^b - \zeta \tag{80}$$

$$P_{\hat{x}\hat{y}} = <\hat{x}^b, \hat{y}^T> = P^b H^T \tag{81}$$

$$P_{\hat{y}\hat{y}} = <\hat{y}, \hat{y}^T> = HP^b H^T + R \tag{82}$$

So, Kalman gain

$$K = P^b H^T (HP^b H^T + R)^{-1} \tag{83}$$

Eq. (83) is identical to Eq. (28). Therefore, Eq. (28), or KF, EKF and EnKF, is a special case of Eq. (76) under the assumption of linear measurement function.

In the standard KF, the state error covariance is updated at each analysis cycle during the measurement update process. Updating the error covariance matrix is important because it represents the change in forecast error covariance when a measurement is performed. The EnKF implementation does not require the covariance update equation because it can directly calculate the updated error covariance matrix from a set of ensemble members. Evensen [17] has derived the analysis of covariance equation that is consistent with the standard KF error covariance to update Eq. (28). But the true representation of the updated error covariance requires a large ensemble size, which is often computationally infeasible.

The general form of the Kalman gain makes use of the reformulated error covariance. In a broad sense, the above algorithm implicitly uses the prior covariance update equation (or the analysis error covariance matrix) to calculate the forecast error covariance. Thus, the above algorithm is fully consistent with the time update and measurement update formulation of the Kalman filter algorithm. On this basis, one can develop a new type of Kalman filter that chooses the ensemble members deterministically in such a way that they can capture the statistical

moments of the nonlinear model accurately. In the next subsection, we will discuss the new type of Kalman filter, called sigma-point Kalman filter, based on the above algorithm.

6. Sigma-point Kalman filters (SPKF)

6.1. Basics of SPKF

EnKF was developed in order to overcome the linearization of nonlinear models. As introduced earlier, the idea behind EnKF is to 'integrate' Fokker-Plank equation using ensemble technique to estimate the forecast error covariance. Theoretically, if the ensemble size is infinite, the estimate approaches the true value. However, in reality, we can only use finite ensemble size, even very small size for many problems, leading to truncation errors. Thus, some concerns exist such as how to wisely generate finite samples for the optimal estimate of prediction error covariance, how much the least ensemble size is for an efficient estimate of error covariance and how much the true error covariance can be taken into account in the EnKF, given an ensemble size. In this section, we will introduce a new ensemble technique for EnKF, which is called sigma-point Kalman filter (SPKF).

The so-called sigma-point approach is based on deterministic sampling of state distribution to calculate the approximate covariance matrices for the standard Kalman filter equations. The family of SPKF algorithms includes the unscented Kalman filter (UKF [26]), the central difference Kalman filter (CDKF [27]) and their square root versions [28]. Another interpretation of the sigma-point approach is that it implicitly performs a statistical linearization of the nonlinear model through a weighted statistical linear regression (WSLR) to calculate the covariance matrices [29]. In SPKF, the model linearization is done through a linear regression between a number of points (called sigma points) drawn from a prior distribution of a random variable rather than through a truncated Taylor series expansion at a single point. It has been found that this linearization is much more accurate than a truncated Taylor series linearization [28]. Eqs. (80)–(82) construct a core of SPKF. A central issue here is how to generate the optimal ensemble members for applying these equations. There are two basic approaches aforementioned, UKF and CDKF. For an L-dimensional dynamical system represented by a set of discretized state-space equations of (67), it has been proven that $2L + 1$ ensemble members, constructed by UKF or CDKF, can precisely estimate the mean and covariance. We ignore the theoretical proof and only outline the UKF scheme as below.

Denote $2L + 1$ sigma points at time k for producing ensemble members by $\chi_k = [\chi_{k,0}, \chi_{k,1}^+, ..., \chi_{k,L}^+, \chi_{k,1}^-, ..., \chi_{k,L}^-]$, which that is defined according to the following expressions:

$$\chi_{k,0} = \bar{X}_k^a \tag{84}$$

$$\chi_{k,i}^{+} = \bar{X}_{k}^{a} + [c\sqrt{P_{X,k}^{a}}]_{i} \tag{85}$$

$$\chi_{k,i}^{-} = \bar{X}_{k}^{a} - [c\sqrt{P_{X,k}^{a}}]_{i} \tag{86}$$
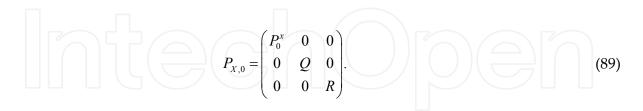
where $L = N_{x} + N_{\eta} + N_{\zeta}$ is the sum of the dimensions of model states, model noise and measurement noise. The augmented state vector $X = [x; \eta; \zeta]$ is a L-dimensional vector. $\sqrt{P_{X,k}^{a}}$ is the covariance of the augmented state vector (analysis) at the previous step. $[\sqrt{P_{X,k}^{a}}]_{i}$ is the ith row (column) of the weighted matrix square root of the covariance matrix (L dimension). c is a scale parameter that will be specified later. The key point here is to produce $(2L + 1)$ ensemble members by integrating model with $2L + 1$ initial conditions of Eqs. (84)–(86); by these ensemble members, the filter Eqs. (80)–(82) will be performed.

The procedure is summarized as below:

1. Initially, perturb a small amount, denoted by $\tilde{x}_{0}$ on initial condition x_{0}, using Evensen method [17]; and also randomly generate perturbation for q and r, drawn from normal distributions of $N(0, Q)$ and $N(0, R)$. Thus, we can construct the augmented state vector and corresponding covariance ($k = 0$)

$$\bar{X}_{0}^{a} = [x_{0}; 0; 0]; \tag{87}$$

$$P_{0}^{x} = \tilde{x}_{0}\tilde{x}_{0}^{T}; \tag{88}$$

$$P_{X,0} = \begin{pmatrix} P_{0}^{x} & 0 & 0 \\ 0 & Q & 0 \\ 0 & 0 & R \end{pmatrix}. \tag{89}$$

2. From the above formula, we can calculate sigma points using Eqs. (84)–(86). Note that each set of sigma points, denoted by χ_{k}, has dimension L, e.g. the ith sigma point can be expressed by $\chi_{k,i} = [x_{k,i}; \eta_{k,i}; \zeta_{k,i}]$.

3. Using the $2L + 1$ sigma points to integrate state-space model. For the ith sigma point, we have $x_{k+1,i}^{f} = f(x_{k,i}, \eta_{k,i})$. When i varies from 1 to $2L + 1$, we produce $2L + 1$ ensemble members, from which analysis mean and covariance will be obtained, which are in turn

used to produce sigma points for next step ($k + 1$), to form a recursive algorithm. Suppose we have $2L + 1$ ensembles, the analysis mean and the covariance are calculated as follows:

$$\bar{x}_{k+1}^f = \sum_{i=0}^{2L} w_i^{(m)} x_{k+1,i}^f \tag{90}$$

$$(P_{xx}^f)_{k+1} = \sum_{i=0}^{2L} w_i^{(c)} [x_{k+1,i}^f - \bar{x}_{k+1}^f][x_{k+1,i}^f - \bar{x}_{k+1}^f]^T \tag{91}$$

$$y_{k+1,i}^f = h\left(x_{k+1,i}^f, \zeta_{k+1,i}\right) \tag{92}$$

$$\bar{y}_{k+1}^f = \sum_{i=0}^{2L} w_i^{(m)} y_{k+1,i}^f \tag{93}$$

$$(P_{yy})_{k+1} = \sum_{i=0}^{2L} w_i^{(c)} [y_{k+1,i}^f - \bar{y}_{k+1}^f][y_{k+1,i}^f - \bar{y}_{k+1}^f]^T \tag{94}$$

$$(P_{xy})_{k+1} = \sum_{i=0}^{2L} w_i^{(c)} [x_{k+1,i}^f - \bar{x}_{k+1}^f][y_{k+1,i}^f - \bar{y}_{k+1}^f]^T \tag{95}$$

$$K_{k+1} = P_{xy} P_{yy}^{-1}, \tag{96}$$

$$\bar{x}_{k+1}^a = \bar{x}_{k+1}^f + K_{k+1}\left[y_{k+1} - \bar{y}_{k+1}^f\right] \tag{97}$$

$$P_{k+1}^a = (P_{xx}^f)_{k+1} - K_{k+1} P_{yy} K_{k+1}^T, \tag{98}$$

where

$$c = \sqrt{L + \lambda} \tag{99}$$

$$w_0^{(m)} = \frac{\lambda}{L + \lambda} \tag{100}$$

$$w_0^{(c)} = \frac{\lambda}{L + \lambda} + 1 - \alpha^2 + \beta \tag{101}$$

$$w_i^{(m)} = w_i^{(c)} = \frac{1}{2(L + \lambda)}, i = 1, 2, \ldots 2L \tag{102}$$

$$\lambda = \alpha^2 (L + \kappa) - L, \tag{103}$$

α and κ are tuning parameters. $0 < \alpha < 1$ and $\kappa \geq 0$. Often κ is chosen 0 as default value and $\beta = 2$.

4. From P_{k+1}^a, as well choosing random perturbation for model noise η and observation noise ζ, drawn from Gaussian distribution of $N(0, Q)$ and $N(0, R)$, we calculate sigma points using Eqs. (84)–(86), and repeat Step 2 and Step 3 and so on until the assimilation is completed for the entire period.

6.2. Remarks of SPKF

SPKF was recently introduced into the earth sciences [15, 30]. The main differences between SPKF and EnKF include

i.   SPKF chooses the ensemble members deterministically while EnKF uses random perturbation to generate ensemble members;

ii.   the number of sigma points is a fixed value as $2L + 1$, while the ensemble size in EnKF is pre-specified;

iii.   SPKF uses Eq. (98) to update the error covariance matrix, while EnKF does not update explicitly the error covariance matrix; and

iv.   Sigma points are calculated using Eqs. (84)–(86) every time when the observation is available, while the ensemble members in EnKF only perturbed in the initial time. Recent application of SPKF on a realistic oceanic model indicates that the SPKF is better than the EnKF in the similar level of computational cost [31].

In SPKF, the number of sigma points is $2L + 1$, here L is the dimension of the augmented state vector $X = [x; \eta; \zeta]$, i.e. $L = N_x + N_\eta + N_\zeta$ is the sum of model state, model noise and observation noise. Usually, L is the order 10^3–10^4, so the computational expense is a huge challenge in SPKF for realistic problems. A solution is to use the truncated singular-value decomposition (TSVD) to reduce the sigma points. As seen from Eqs. (84)–(86), the $P_{X,k}^a$ is a $L * L$ matrix, thus the dimension of $P_{X,k}^a$ determines the ensemble size. Suppose that $P_{X,k}^a$ can be expressed as

$$P_{X,k}^a = E_{X,k}^a \Sigma_k (E_{X,k}^a)^T \tag{104}$$

where $\Sigma_k = \mathrm{diag}(\sigma_k^1, \sigma_k^2, ..., \sigma_k^L)$ is a diagonal matrix of eigenvalues that are sorted in descending order, i.e. $\sigma_k^1 \geq \sigma_k^2 \geq ... \geq \sigma_k^L$, and $E_{X,k}^a = [e_{X,k,1}^a, e_{X,k,2}^a, ..., e_{X,k,L}^a]$. Truncating the first m modes, so we can write the sigma points (84)–(86) as below:

$$\chi_{k,0} = \bar{X}_k^a \tag{105}$$

$$\chi_{k,i}^+ = \bar{X}_k^a + c\sqrt{\sigma_k^i}\, e_{X,k,i}^a \tag{106}$$

$$\chi_{k,i}^- = \bar{X}_k^a - c\sqrt{\sigma_k^i}\, e_{X,k,i}^a \tag{107}$$

$i = 1, 2, ..., m$. Thus, the ensemble size becomes $2 * m + 1$, where $m < < L$. Some fast SVD algorithms can be used here, such as Lanczos and block Lanczos [32]. The application of the truncated SVD was also found in [33, 34].

Further simplifying $P_{X,k}^a$ based on its definition (or Cholesky decomposition), i.e. $P_{X,k}^a = A_{X,k}^a * (A_{X,k}^a)^T$, where $A_{X,k}^a$ is the data that has subtracted the ensemble mean. Thus, Eqs.(82)–(84) can be written as follows:

$$\chi_{k,0} = \bar{X}_k^a \tag{108}$$

$$\chi_{k,i}^+ = \bar{X}_k^a + [cA_{X,k}^a]_i \tag{109}$$

$$\chi_{k,i}^- = \bar{X}_k^a - [cA_{X,k}^a]_i \tag{110}$$

where $[cA_{X,k}^a]_i = [x_k^a; \eta_k; \zeta_k]_i, i = 1, 2, ..., L$, $(x_k^a)_i = (x_k^f)_i + K_k[y_k - y_k^f]$. Eqs.(109) and (110) transfer the covariance matrix $P_{X,k}^a$ to data matrix $A_{X,k}^a$ in constructing sigma points. The largest advantage is to avoid explicit expression of $P_{X,k}^a$, which could be a very large matrix beyond memory of current computers. However, Eqs.(109) and (110) cannot reduce the ensemble size $(2L + 1)$. A solution is to decompose, such as principal component analysis, as

used in [14]. Further discussions on optimal construction of sigma points should be conducted for a realistic application of SPKF.

Again, we look at sigma-point generation, i.e. Eqs. (106) and (107) or (109) and (110). As we defined, an augmented matrix is applied here $[x; \eta; \zeta]$. Without losing the generality, rewrite them as below:

$$\begin{bmatrix} x_{k,0} \\ \eta_{k,0} \\ \zeta_{k,0} \end{bmatrix} = \begin{bmatrix} \overline{x}_{k,0} \\ 0 \\ 0 \end{bmatrix} \tag{111}$$

$$\begin{bmatrix} x_{k,i} \\ \eta_{k,i} \\ \zeta_{k,i} \end{bmatrix} = \begin{bmatrix} \overline{x}_{k,0} \\ 0 \\ 0 \end{bmatrix} + c \begin{bmatrix} x_{k,i}^{a} \\ \eta_{k,i} \\ \zeta_{k,i} \end{bmatrix} \tag{112}$$

Similarly, we can write Eq. (107) or (110) using individual variables. From Eqs. (111) and (112), we can draw

- Noise and model state analyses in constructing sigma points at k step are independent. It should be noted that x_k^a is from Eq. (97) and noise are draw from a Gaussian distribution. If we assume that noise is taken randomly each time, x_k^a is only relevant to noise that is drawn at time step k, and independent with model noise and observation noise drawn for analysis of the time step $k + 1$, thus, $P_{X,k}$ is a diagonal block matrix, i.e.

$$P_{X,k} = \begin{pmatrix} P_k^x & 0 & 0 \\ 0 & Q & 0 \\ 0 & 0 & R \end{pmatrix} \tag{113}$$

- There are no update equations for noise, so they are randomly taken from Gaussian distribution, i.e. the index i in η_i and ζ_i actually does not have meaning. Thus, it should be a reasonable assumption that the η_i and ζ_i, used for constructing sigma points at time step $k + 1$, are not related to $P_{X,k}$ (time step of k), as argued above. Thus, Eq. (108) always holds unless the noise is designed considering the temporal coherence such as red noise in time.

- Based on the above, the actual ensemble size is $2N_x + 1$, and not $2L + 1$. This is because neither model noise nor observation noise can produce ensemble alone. Model errors η_i and $x_{k,i}^f$ must be joined together to produce ensemble members with N_x. Let us see this in details:

at the initial time, initial perturbation on model states plus drawn noise for model errors and measurement errors are with mean and variance as follows:

$$\bar{X}_0^a = \begin{bmatrix} x_0; 0; 0 \end{bmatrix}, \quad P_{X,0} = \begin{pmatrix} P_0^x & 0 & 0 \\ 0 & Q & 0 \\ 0 & 0 & R \end{pmatrix} \tag{114}$$

Theoretically, there are $2(N_x + N_\eta + N_\zeta) + 1$ ensembles, denoted by the ith column of $P_{X,0}$ ($i = 1, ..., N_x$; $Nx + 1, ..., N_x + N_\eta$; $N_x + N_\eta + 1, ..., N_x + N_\eta + N_\zeta$) and formula (84)–(86). However, at the ith column, the elements of the row, indicating the model inputs (x, η, ζ), only have the non-zero values of N_x. Obviously, the sigma points of zero-values makes the update equation $\chi_{k+1,i} = f(\chi_{k,i})$ invalid, thus, the actual ensemble size is $2N_x + 1$.

When truncation technique is applied to reduce the ensemble size, the ensemble spread might be shrunk due to relatively small ensemble size. Like EnKF, an inflation approach of SPKF might be helpful. It is interested in developing such a scheme for SPKF. Also, we can localize SPKF, like localized EnKF, to solve memory and computation issues.

All of the remarks of SPKF are from the authors' thinking and understanding. It is interesting to further test and validate these ideas and properties using simple models.

7. Beyond Kalman filters: particle filter and its derivatives

7.1. Standard particle filter

We have introduced the Kalman filter (KF), extended Kalman filter (EKF), ensemble Kalman filter (EnKF) and sigma-point Kalman filter (SPKF) in previous sections. All of those filters belong to the sequential data assimilation method, i.e. observation data is assimilated into the model system as soon as it is available. The Bayesian estimation theory provides a general framework of the sequential data assimilation methods. If we assume the state-space model is given by Eqs. (34) and (35), the analysis step of a Bayesian-based assimilation method is deduced by Bayes' theorem:

$$p(x_t|y_t) = \frac{p(y_t|x_t) p(x_t)}{p(y_t)}, \tag{115}$$

where $p(y_t)$ plays as a normalization factor.

Recalling Section 2.3, Eq. (12) actually assumes that the prior probability density function $p(x_t)$ and the likelihood function $p(y_t|x_t)$ are Gaussian distribution functions, and thus the posterior

probability density function $p(x_t|y_t)$ is also a Gaussian. Based on the Gaussian assumption, the cost function of 3D-Var (i.e. Eq. (6)) can be derived, and it is equivalent to the Kalman filter Eqs. (27)–(29). All the Kalman-based filters (e.g. EKF, EnKF, EnSRF, SPKF, etc.) contain the inherent Gaussian assumption, and they are derived and validated for Gaussian systems in theory. However, this Gaussian assumption is often not applicable for nonlinear systems. Even for an initial Gaussian error, it often becomes non-Gaussian while propagating forward with nonlinear models.

The particle filter (PF) is a sequential data assimilation method that is able to deal with the nonlinear and non-Gaussian state estimation problem. Like EnKF, PF also uses an ensemble, but it is used to approximately estimate the full probability density function rather than only the error covariance B. An ensemble member is also referred to as a particle in PF literatures. Suppose the prior probability density is the sum of Dirac delta functions

$$p(x_t) = \sum_{i=1}^{N} \delta(x_t - x_t^i) \tag{116}$$

where $\{x_t^i, i = 1, 2, ..., N\}$ are particles drawn from $p(x_t)$. The posterior probability density is derived by applying the Bayes' theorem directly, that is

$$p(x_t|y_t) \propto p(y_t|x_t)p(x_t) = \sum_{i=1}^{N} w_{t,i} \delta(x_t - x_t^i) \tag{117}$$

in which $w_{t,i} \propto p(y_t|x_t^i)$, and a normalization step, is required to make $\{w_{t,i}, i = 1, 2, ..., N\}$ sum up to 1. If we assume the likelihood function is Gaussian, $w_{t,i}$ can be computed by

$$p(y_t | x_t^i) = \frac{1}{\sqrt{2\pi R}} \exp\{-1/2[y_t - h(x_t^i)]R^{-1}[y_t - h(x_t^i)]^T\}. \tag{118}$$

Or else we can use any specified probability density function of $p(y_t|x_t)$ to compute the likelihood.

With the posterior probability density function $p(x_t|y_t)$, the analysis value and covariance can be computed by

$$\bar{x}_t = \int x * p(x|y_t)dx = \sum_{i=1}^{N} w_{t,i} x_t^i \tag{119}$$

$$\mathrm{var}(x_t) = \int x^2 * p(x|y_t)dx - \bar{x}_t^2 = \sum_{i=1}^{N} w_{t,i}(x_t^i)^2 - \bar{x}_t^2 \tag{120}$$

and higher-order moments of the posterior state can also be estimated.

Before stepping forward to next stage, a resampling step is required to make each particle with uniform weight. A typical resampling strategy is the sequential importance resampling (SIR) that removes particles with very small weights and duplicates those with large weights. A detailed algorithm of SIR can be found in [35]. The resampling algorithm gives the indices and number of copies of those particles that should be duplicated, i.e. computes $s_1, s_2, ..., s_N$ according to the weights, where each $s_i \in 1, 2, ..., N$. And then $\left\{ x_t^{s_i}, i = 1, 2, ..., N \right\}$ are regarded as new particles.

In summary, the algorithm of standard particle filter is given below:

1. generate the initial ensemble $\left\{ x_0^i, i = 1, 2, ..., N \right\}$ as EnKF does;

2. integrate the model until the observation is available;

3. use Eq. (118) to compute the weight for each particle, and normalize them;

4. use Eq. (119) to obtain the analysis and Eq. (120) to obtain the covariance if necessary;

5. apply the resampling algorithm to derive the resampling indices, and derive the new ensemble $\left\{ x_t^{s_i}, i = 1, 2, ..., N \right\}$; and

6. repeat Steps 2–5 until the end of assimilation period.

The standard particle filter [36] is also known as the bootstrap particle filter or SIR particle filter.

7.2. Variants of PF

The particle filter is a highly promising technique because it does not invoke any Gaussian assumptions. It has been widely used and studied in many other fields. The PF estimates the full probability density function of the forecasted state based on an ensemble of states with different weights. However, the PF suffers from the problem of filter degeneracy, i.e. the procedure collapses to a very small number of highly weighted particles among a horde of almost useless particles carrying a tiny proportion of the probability mass. Even if resampling techniques are used, the degeneracy cannot be completely avoided with limited ensemble size. The number of particles must grow substantially with the dimension of the system to avoid degeneracy [37, 38], a requirement that is apparently too costly for large models such as GCMs. Various efforts have been made to resolve this issue, as documented in an excellent overview [39].

Several strategies are often employed to address the problem of filter degeneracy in applications of the particle filter. For example, Papadakis et al. proposed a weighted ensemble Kalman filter (WEnKF) [40] that uses an ensemble-based Kalman filter as the proposal density from which the particles are drawn. Van Leeuwen et al. developed a fully nonlinear particle filter

by exploiting the freedom of the proposal transition density, which ensures not only that all particles ultimately occupy high-probability regions of state-space but also that most of the particles have similar weights [41]. The implicit particle filter uses gradient descent minimization combined with random maps to find the region of high probability, avoiding the calculation of Hessians [42]. Luo et al. have proposed an efficient particle filter that uses residual nudging to prevent the residual norm of the state estimates from exceeding a pre-specified threshold [43]. These particle filters were very recently proposed and have attracted broad attention in the community of atmos./ocean. data assimilation. Below, we will briefly introduce the equivalent weights particle filter (EWPF) by Van Leeuwen [39, 41].

The equivalent weights particle filter is a fully nonlinear data assimilation method that works in a two-stage process. It uses the proposal density to ensure that the particles have almost equivalent weights, by which the filter degeneracy can be avoided.

In the standard PF, the particles at time step t are propagated by the original model, i.e. $x_{t+1}^i = f(x_t^i) + \eta_t$, which implies that the particles at time step $t + 1$ are drawn from the transition density $p(x_{t+1} | x_t)$. In that case, the weight of each x_{t+1}^i varies greatly and filter degeneracy is very likely to happen.

In EWPF, another transition density, call the proposal density, is introduced. The proposal density depends on the future observation y_{t+1} and all previous particles $\{x_t^i, i = 1, 2, ..., N\}$. With the help of proposal density, the particle x_t^i is propagated using a different model

$$x_{t+1}^i = g\left(x_t^i, y_{t+1}\right) + \eta_t. \tag{121}$$

The model g can be anything, for instance, one can add a relaxation term and change random forcing:

$$x_{k+1}^i = f\left(x_k^i\right) + \eta_k^i + A\left(y_{t+1} - H\left(x_k^i\right)\right), \quad k = 1, ..., p(k) \tag{122}$$

where $p(k)$ is a function of the time between observations, and each k implies each model step without observation. A is a relaxation term that will 'drag' the particle towards future observation. In [44], it is given by

$$A = p(k)QH^T R^{-1}, \tag{123}$$

where the matrices Q and R correspond to the model error covariance and observation error covariance, respectively.

The second stage of EWPF involves updating each particle at the observation time $t + 1$ via the formula

$$x_{t+1}^i = f(x_t^i) + \alpha_i QH^T (HQH^T + R)^{-1}(y_{t+1} - H(f(x_t^i))) + \eta_t^i \tag{124}$$

where α_i are scalers computed so as to make the weights of the particles equal. Using the expression for weights and setting all weights equal to a target weight (e.g. $1/N$)

$$w_i = p\left(y_{t+1}|x_{t+1}^i(\alpha_i)\right) = w_{\text{target}} \tag{125}$$

α_i can be solved by numerical methods.

Eqs. (122)–(125) show an example of how to construct the proposal model g in(121)), it can also be done by running 4D-var on each particle (implicit particle filter), or using the EnKF as proposal density. Those methods refer to Morzfeld et al. [42] and Papadakis et al. [40].

7.3. Remarks of PF

7.3.1. Combined method of EnKF and PF

The ensemble Kalman particle filter (EnKPF) is a combination of the EnKF and the SIR particle filter. It was recently introduced to address non-Gaussian features in data assimilation for highly nonlinear systems, by providing a continuous interpolation between the EnKF and SIR-PF analysis schemes [45].

As stated above, both EnKF and PF methods are based on the Bayesian estimation theory, but they approximate the probability density function of the state in different ways. The EnKF only approximates the mean and covariance of the state through a series of equally weighted ensemble members. And the particle filter considers the weights of the ensemble members according to the likelihoods. The EnKF contains the Gaussian assumption but requires relatively small ensemble size to prevent filter degeneracy, which is in contrast with the PF.

The EnKPF takes advantage of both methods by combining the analysis schemes of the EnKF and the SIR-PF using a controllable index (i.e. tuning parameter). In contrast with both the EnKF and the SIR-PF, the analysis scheme of the EnKPF not only updates the ensemble members but also considers the weights.

Assume that the forecast ensemble $\left\{x_i^f, i = 1, 2, ..., N\right\}$ and the observation data y are available, and that the forecast covariance P^f can be calculated using the ensemble, the analysis scheme of EnKPF is given below.

1. Choose $\gamma \in [0, 1]$ and apply the EnKF that is based on the inflated observation error covariance R/γ as follows:

$$K_1(\gamma) = P^f H^T (HP^f H^T + R/\gamma)^{-1} = \gamma P^f H^T (\gamma HP^f H^T + R)^{-1} \tag{126}$$

$$v_i = x_i^f + K_1(\gamma)(y - Hx_i^f) \tag{127}$$

$$Q = \frac{1}{\gamma} K_1(\gamma) R K_1(\gamma)^T \tag{128}$$

2. Compute the weights w_i for each updated member v_i as follows:

$$w_i = \phi\left(y; Hv_i, \frac{R}{1-\gamma} + HQH^T\right) \tag{129}$$

and normalize the weights by $\widehat{w}_i = w_i / \sum_{i=1}^{N} w_{i'}$ in which φ is the probability density function of a Gaussian.

3. Calculate the resampling index $s(i)$ for each member v_i according to $\widehat{w}_i$ using the SIR algorithm, then set

$$x_i^u = v_{s(i)} + K_1(\gamma) \frac{\epsilon_{1,i}}{\sqrt{\gamma}} \tag{130}$$

where $\epsilon_{1,i}$ is a random observation error drawn from the Gaussian $N(0, R)$.

4. Compute $K_2(1-\gamma) = (1-\gamma) QH^T [(1-\gamma) HQH^T + R]^{-1}$, and generate $\epsilon_{2,i}$ from $N(0, R)$ and EnKF with the inflated observation error again as follows:

$$x_i^a = x_i^u + K_2(1-\gamma)\left[y + \frac{\epsilon_{2,i}}{\sqrt{1-\gamma}} - Hx_i^u\right] \tag{131}$$

γ can be determined recursively to match the optimal performance of EnKPF. More details of EnKPF can be found in [45, 46].

7.3.2. Localization in PF

Previous sections have introduced the localization technique in EnKF, which greatly improves the performance of EnKF in high-dimensional models. The advantages of localization motivate the search for a localization procedure in particle filtering.

Van Leeuwen had a deep discussion on this topic [39]. He argued that *one can calculate the weights locally, but it is not easy for resampling. In the resampling step low-weight particles are abandoned and high-weight particles are duplicated. However, with local weights, different particles are selected in different parts of the domain. The problem is that we have to have continuous (in space) model fields to propagate forward in time with the model. Just constructing a new particle that consists of one particle in one part of the model domain and another particle in another domain will lead to problems at the boundary between these two.*

The problem of spatial discontinuity makes the localization in particle filter not feasible currently. Most of the advanced particle filters (e.g. EWPFand implicit particle filter) are using the idea of global weight, i.e. the weight for each member is a scalar.

However, there are still some attempts on the localization in particle filter. For example, Poterjoy developed the localized particle filter (LPF) that updates particles locally using ideas borrowed from EnKF [47]. The paper has demonstrated some advantages of the new filter over EnKF, especially when the observation networks consist of densely spaced measurements that relate nonlinearly to the model state. This is a very interesting work about the particle filter, it also has a potential to work with large atmos./ocean. data assimilation systems.

8. Remarks and conclusions

Data assimilation is the process by which observations of the actual system are incorporated into a numerical model to optimally estimate the system states. In this chapter, we introduced several ensemble-based data assimilation methods that are widely used in the earth sciences. One can read it as an introduction to ensemble-based data assimilation methods, but also can view it as a brief review of the application of these ensemble-based assimilation methods on the earth sciences. It is author's effort to write such a 'review' chapter with introductory language, making it more readable. As found in the chapter, many discussions, derivations and analyses are actually very thoughtful, not only introducing these methods, but also deepening the understanding to them. This is emphasized by the analysis of the rationale behind each method, including: i). the principle for deriving the algorithm; ii) basic assumptions of each method; iii). the connection and relation of different methods (e.g., EKF and EnKF, EnKF and SPKF etc.); iv). the advantages and deficiencies of each method. Especially we put rather weights to discuss potential concerns, challenges and possible solutions when these methods are applied to high-dimensional systems in the earth sciences. This chapter can be a "textbook" for the beginners to learn these data assimilation algorithms, and also a good reference for researchers for better understanding and applying these methods.

Acknowledgements

Author details

Youmin Tang[1,2*], Zheqi Shen[2] and Yanqiu Gao[2]

*Address all correspondence to: ytang@unbc.ca

1 Environmental Science and Engineering, University of Northern British Columbia, Prince George, British Columbia, Canada

2 State Key Laboratory of Satellite Ocean Environment Dynamics, Second Institute of Oceanography, State Oceanic Administration, Hangzhou, China

References

[1] Talagrand O. Assimilation of observations, an introduction. Journal of Meteorological Society of Japan Series 2. 1997;75:81–99.

[2] Kalnay E. Atmospheric modeling, data assimilation, and predictability. Cambridge University Press. New York. 2003.

[3] Le Dimet F. X., Talagrand O. Variational algorithms for analysis and assimilation of meteorological observations: Theoretical Aspects. Tellus A. 1986;38(2):97–110.

[4] Courtier P., Andersson E., Heckley W., Vasiljevic D., Hamrud M., Hollingsworth A. The ECMWF implementation of three-dimensional variational assimilation (3D-Var). I:Formulation. Quarterly Journal of the Royal Meteorological Society. 1998;124(550): 1783–1807.

[5] Kalman R. E. A new approach to linear filtering and prediction problems. Journal of Basic Engineering. 1960;82(1):35–45.

[6] Rabier F., Jarvinen H., Klinker E., Mahfouf J. F., Simmons A. The ECMWF operational implementation of four-dimensional variational assimilation. I: Experimental results with simplified physics. Quarterly Journal of the Royal Meteorological Society. 2000;126(564):1143–1170.

[7] Mahfouf J. F., Rabier F. The ECMWF operational implementation of four-dimensional variational assimilation. II: Experimental results with improved physics. Quarterly Journal of the Royal Meteorological Society. 2000;126(564):1171–1190.

[8] Klinker E., Rabier F., Kelly G., Mahfouf J. F. The ECMWF operational implementation of four-dimensional variational assimilation. III: Experimental results and diagnostics

with operational configuration. Quarterly Journal of the Royal Meteorological Society. 2000;126(564):1191–1215.

[9] Houtekamer P. L., Mitchell H. L., Pellerin G., Buehner M., Charron M., Spacek L., et al. Atmospheric data assimilation with an ensemble Kalman filter: Results with real observations. Monthly Weather Review. 2005;133(3):604–620.

[10] Evensen G. Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. Journal of Geophysical Research: Oceans (1978–2012). 1994;99(C5):10143–10162.

[11] Houtekamer P. L., Mitchell H. L. Data assimilation using an ensemble Kalman filter technique. Monthly Weather Review. 1998;126(3):796–811.

[12] Evensen G. Advanced data assimilation for strongly nonlinear dynamics. Monthly Weather Review. 1997;125(6):1342–1354.

[13] Keppenne C. L. Data assimilation into a primitive-equation model with a parallel ensemble Kalman filter. Monthly Weather Review. 2000;128:1971–1981.

[14] Houtekamer P. L., Mitchell H. L. A sequential ensemble Kalman filter for atmospheric data assimilation. Monthly Weather Review. 2001;129(1):123–137.

[15] Ambadan J. T., Tang Y. Sigma-point Kalman filter data assimilation methods for strongly. Journal of the Atmospheric Sciences. 2009;66(2):261–285.

[16] Tang Y., Ambandan J., Chen D. Nonlinear measurement function in the ensemble Kalman filter. Advances in Atmospheric Sciences. 2014;31(3):551–558.

[17] Evensen G. The ensemble Kalman filter: Theoretical formulation and practical implementation. Ocean Dynamics. 2003;53(4):343–367.

[18] Burgers G., van Leeuwen P. J., Evensen G. Analysis scheme in the ensemble Kalman filter. Monthly Weather Review. 1998;126(6):1719–1724.

[19] Whitaker J. S., Hamill T. M. Ensemble data assimilation without perturbed observations. Monthly Weather Review. 2002;130(7):1913–1924.

[20] Tippett M. K., Anderson J. L., Bishop C. H., Hamill T. M., Whitaker J. S. Ensemble square root filters. Monthly Weather Review. 2003;131(7):1485–1490.

[21] Bishop C. H., Etherton B. J., Majumdar S. J. Adaptive sampling with the ensemble transform Kalman filter. Part I: Theoretical aspects. Monthly Weather Review. 2001;129(3):420–436.

[22] Anderson J. L. An ensemble adjustment Kalman filter for data assimilation. Monthly Weather Review. 2001;129(12):2884–2903.

[23] Hunt B. R., Kostelich E. J., Szunyogh I. Efficient data assimilation for spatiotemporal chaos: A local ensemble transform Kalman filter. Physica D: Nonlinear Phenomena. 2007;230(1):112–126.

[24] Gaspari G., Cohn, S. Construction of correlation functions in two and three dimensions. Quarterly Journal of the Royal Meteorological Society. 1999;125(554):723–757.

[25] Anderson J. L. Ensemble Kalman filters for large geophysical applications. IEEE Control Systems. 2009;29(3):66–82.

[26] Julier S. J., Uhlmann J. K., Durrant-Whyte H. F. A new approach for filtering nonlinear systems. Proceedings of American Control Conference. 1995;3:1628–1632.

[27] Ito K., Xiong K. Gaussian filters for nonlinear filtering problems. IEEE Transactions on Automatic Control. 2000;45(5):910–927.

[28] van Der Merwe R., Wan E. Efficient derivative-free Kalman filters for online learning. In: Proceedings of ESANN. 2001.

[29] Gelb A. Applied optimal estimation. MIT Press. Cambridge. 1974.

[30] Luo X., Morez I. M. Ensemble Kalman filter with the unscented transform. Physica D: Nonlinear Phenomena. 2009;238(5):549–562.

[31] Tang Y., Deng Z., Manoj K. K., Chen D. A practical scheme of the sigma-point Kalman filter for high-dimensional systems. Journal of Advances in Modeling Earth Systems. 2014;6:21–37.

[32] Golub G. H., van Loan C. F. Matrix computations. JHU Press. Maryland. 2012.

[33] Hansen B. Atmospheric data assimilation with an ensemble Kalman filter: Results with real observations. Monthly Weather Review.2005;133(3):604–620.

[34] Ehrendorfer M., Tribbia J. J. Optimal prediction of forecast error covariance through singular vectors. Journal of the Atmospheric Sciences. 1997;52(2):286–313.

[35] Arulampalam M. S., Maskell S., Gordon N., Clapp T. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. IEEE Transactions on Signal Processing. 2002;50(2):174–188.

[36] Gordon N. J., Salmond D. J., Smith A. F. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. IEE Proceedings F (Radar and Signal Processing). 1993;140(2):107–113.

[37] Bengtsson T., Snyder C., Nychka D. Toward a nonlinear ensemble filter for high-dimensional systems. Journal of Geophysical Research. 2003;108(D24):8775.

[38] Snyder C., Bengtsson T., Bickel P., Anderson J. Obstacles to high-dimensional particle filtering. Monthly Weather Review. 2008;136(12):4629–4640.

[39] van Leeuwen P. J. Particle filtering in geophysical systems. Monthly Weather Review. 2009;137(12):4089–4114.

[40] Papadakis N., Mémin E., Cuzol A., Gengembre N. Data assimilation with the weighted ensemble Kalman filter. Tellus A. 2010;62(5):673–697.

[41] van Leeuwen P. J. Efficient nonlinear data-assimilation in geophysical fluid dynamics. Computers and Fluids. 2011;46(1):52–58.

[42] Morzfeld M., Chorin A. J. Implicit particle filtering for models with partial noise, and an application to geomagnetic data assimilation. Nonlinear Processes in Geophysics. 2012;19(3):365–382.

[43] Luo X., Hoteit I. Efficient particle filtering through residual nudging. Quarterly Journal of the Royal Meteorological Society. 2014;140(679):557–572.

[44] Browne P. A., van Leeuwen P. J. Twin experiments with the equivalent weights particle filter and HADCM3. Quarterly Journal of the Royal Meteorological Society. 2015;141(693):3399–3414.

[45] Frei M., Kunsch H. R. Bridging the ensemble Kalman and particle filters. Biometrika. 2013;100(4):781–800.

[46] Shen Z., Tang Y. A modified ensemble Kalman particle filter for non-Gaussian systems with nonlinear measurement functions. Journal of Advances in Modeling Earth Systems. 2015;7(1):50–66.

[47] Poterjoy J. A localized particle filter for high-dimensional nonlinear systems. Monthly Weather Review. 2016;144(1):59–76.