*Tellus (2007), 59A, 749–757*
*Printed in Singapore. All rights reserved*



**TELLUS**

# Cluster ensemble Kalman filter

*By* KESTON W. SMITH*; *Woods Hole Oceanographic Institute, Woods Hole, MA, USA*

(Manuscript received 6 November 2006; in final form 1 March 2007)

ABSTRACT

A modified ensemble Kalman filter (KF) is proposed which can enhance performance for highly non-linear prognostic models. The algorithm differs from the traditional ensemble KF by the addition of an expectation maximization step, which estimates the parameters of a Gaussian mixture model for the ensemble of forecast states. The algorithm is tested in twin experiments using a simple phytoplankton–zooplankton model.

## 1. Background and motivation

The Kalman filter (KF) is a maximum likelihood estimator if the model error and measurement error distribution are Gaussian. For non-linear dynamical models the Gaussian assumption is not generally valid, and the KF update will not give a maximum likelihood update. The degree to which the forecast distribution strays from a Gaussian distribution depends on the model dynamics as well as the length of the forecast. The ensemble Kalman filter (EnKF) is an approximate KF in which the forecast of the model error covariance matrix is accomplished by a finite set of Monte Carlo model solutions.

In addition to the EnKF other Monte Carlo data assimilation methods such as sequential importance resampling filter (SIRF) have been applied to high resolution ocean models (van Leeuwen, 2003). The SIRF overcomes the assumption of Gaussian error statistics made in the KF and EnKF by resampling the forecast ensemble based only on the likelihood of each ensemble member. The method is appropriate for all non-linear models however in its usual formulation the SIRF requires larger ensembles than the EnKF. Here, I present a filtering method that is appropriate for stochastically forced non-linear models with error distributions that can be reasonably approximated by a Gaussian mixture model (GMM). A GMM is a model of a random process who's probability density function (pdf) is a weighted sum of a finite number, $n_c$, Gaussian pdfs. That is, if $X$ is a random variable drawn from a GMM and $A$ is some region in $\Re^n$ then

$$P(X \in A) = \int_A \sum_{k=1}^{n_c} \tau_k \frac{\exp\left[-\frac{1}{2}(x-\mu_k)^T {\boldsymbol{\Sigma}_k}^{-1}(x-\mu_k)\right]}{\sqrt{(2\pi)^n |\boldsymbol{\Sigma}_k|}}\,\mathrm{d}x, \tag{1}$$

where $\tau_k$ is the probability that $X$ is drawn from the $k$th component distribution and $\mu_k$ and $\boldsymbol{\Sigma}_\mathbf{k}$ are the mean and covariance of the $k\{\text{th}\}$ component distribution. I propose a new sequential data assimilation scheme, the cluster ensemble Kalman filter (CEnKF), in which the analysis step is proceeded by a cluster analysis (CA) step. In the cluster analysis step a GMM is estimated from the forecast ensemble of states. In the analysis step a Kalman gain matrix is computed for each component distribution in the GMM. The analysis ensemble is then remapped according to the likelihoods of the component distributions, and their relative association with each ensemble member.

The CEnKF is demonstrated using a stochastically forced version of the phytoplankton–zooplankton model of Steele and Henderson (Steele and Henderson, 1992, hereafter SH92). This model was chosen because the covariance between the phytoplankton and zooplankton variables change drastically over state space. This characteristic is common to many non-linear stochastic differential equations, however the SH92 model has only two dimensions so it is relatively easy to visualize.

### 1.1. Kalman filter

The KF is a sequential application of a minimum error variance linear estimator. When data are available the forecast state is updated to obtain the analysis state by

$$\psi^a = \psi^f + \mathbf{P^f H}^T(\mathbf{HP^f H}^T + \mathbf{R})^{-1}(d - \mathbf{H}\psi^f). \tag{2}$$

The analysis covariance is thus,

$$\mathbf{P^a} = \mathbf{P^f} - \mathbf{P^f H}^T(\mathbf{HP^f H}^T + \mathbf{R})^{-1}\mathbf{HP^f}. \tag{3}$$

A summary of notation used here is found in Table 1. Between observation times the model and error covariance are advanced forward in time by a linear dynamical model

$$\psi^f = \mathbf{F}\psi^a, \tag{4}$$

*Correspondence.
e-mail: kwsmith@whoi.edu
DOI: 10.1111/j.1600-0870.2007.00246.x

*Table 1.* Notation used throughout this document

| |
|---|
| $\psi$ model state |
| $\psi^f$ forecast model state |
| $\psi^a$ analysis model state |
| $\psi^t$ true state of the system |
| $\mathbf{H}$ linear measurement operator |
| $d$ measured data, $d = \mathbf{H}\psi^t + e$ |
| $e$ measurement error, assumed to have mean $\mathbf{0}$ and covariance $\mathbf{R}$ |
| $\mathbf{F}$ prognostic model operator, $\psi(t+s) = \mathbf{F}(t, s)\psi(t)$ |
| $\mathbf{P} = E[(\psi - \psi^t)(\psi - \psi^t)^T]$ is the model error covariance |
| $\mathbf{P^f} = E[(\psi^f - \psi^t)(\psi^f - \psi^t)^T]$ is the forecast error covariance |

$$\mathbf{P^f} = \mathbf{F}\mathbf{P^a}\mathbf{F}^T + \mathbf{Q}, \tag{5}$$

where $\mathbf{Q} = E[qq^T]$ and $q$ is the stochastic forcing associated with $\mathbf{F}$. By construction there is no way to directly apply the KF to a general non-linear model $f$, $\psi^f = f(\psi^a, q)$. Additionally, the advancement of the full error covariance matrix, $\mathbf{F}\,\mathbf{P^a}\mathbf{F^T}$, is prohibitive for realistic ocean and atmospheric circulation models for which the dimension of the state space is very large.

### 1.2. Ensemble Kalman filter

The EnKF was proposed by Evensen (Evensen, 1994). It has been applied to state estimation problems in oceanography, meteorology and ecology (Eknes and Evensen, 2002; Evensen, 2003). The EnKF's utility, in data assimilation with non-linear models is owed to its mathematical elegance, ease of implementation, computational efficiency and the absence of the need for derivative calculation (Evensen, 2003, 2006).

A finite number of model states, $n_e$, are simulated from a stochastically forced dynamic model to create an ensemble forecast of states $\{\psi_i^f\} = \{\mathbf{F}(\psi_i(t_0)) + q_i\}$, where the $q_i$ are Monte Carlo simulations of the stochastic forcing. The ensemble prediction is run forward until a time when observations are available.

At the times when data are available the error covariance matrix $P^f$ is approximated by the covariance of the forecast ensemble,

$$\mathbf{P_e^f} = \overline{(\psi^f - \overline{\psi^f})(\psi^f - \overline{\psi^f})^T}, \tag{6}$$

where $\overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$ denotes the ensemble average of the variable $x$. In practice only the terms that are actually needed for the Kalman gain matrix are calculated,

$$\mathbf{H}\mathbf{P_e^f}\mathbf{H^T} = \overline{(\mathbf{H}\psi^f - \overline{\mathbf{H}\psi^\mathbf{f}})(\mathbf{H}\psi^f - \overline{\mathbf{H}\psi^f})^T}, \tag{7}$$

and

$$\mathbf{P_e^f}\mathbf{H^T} = \overline{(\psi^f - \overline{\psi^f})(\mathbf{H}\psi^f - \overline{\mathbf{H}\psi^f})^T}. \tag{8}$$

For each ensemble state a measurement noise vector is simulated from the measurement error model, $e_j \sim G(0, R)$ where $G(0, R)$ is the Gaussian distribution with mean 0 and covariance $\mathbf{R}$. The forecast ensemble is then updated:

$$\psi_i^a = \psi_i^f + \mathbf{P_e^f}\mathbf{H^T}(\mathbf{H}\mathbf{P_e^f}\mathbf{H^T} + \mathbf{R})^{-1}\left(\mathbf{d} - \mathbf{H}\psi_i^f - e_j\right). \tag{9}$$

The ensemble of analysis model states can then be integrated forward in time until the time of the next observations.

Although the EnKF can be applied in a straightforward way to non-linear models the error estimate will not be optimal, because of the non-Gaussian error distribution, regardless of ensemble size.

### 1.3. Cluster analysis

Cluster analysis is, broadly, the automatic search for subgroups of a data set (Fraley and Raftery, 2002). Cluster analysis has been applied widely in biology, data mining and other fields. Approaches to CA vary from ad hoc procedures to the more formal expectation maximization approach taken here. I use cluster analysis to identify a GMM describing the forecast ensemble. Note that this is carried out before the acquisition of the data being assimilated. Expectation maximization (EM) is chosen here for the CA algorithm because it is flexible and consistent with the assumption of a GMM.

## 2. Cluster ensemble Kalman filter

### 2.1. Cluster analysis and expectation maximization

*Gaussian mixture models.* In the CEnKF I assume the prior model at the analysis step is a GMM. For any forecast state $\psi$,

$$p(\psi|\mu_1, \mu_2, \ldots, \mu_{n_c}, \mathbf{\Sigma}_1, \Sigma_2, \ldots, \Sigma_{n_c}, \tau_1, \tau_2, \ldots, \tau_{n_c}) = \sum_{k=1}^{n_c} \tau_k\, p(\psi|\mu_k, \mathbf{\Sigma}_k), \tag{10}$$

$$= \sum_{k=1}^{n_c} \tau_k \frac{\exp\left[-\frac{1}{2}(\psi - \mu_k)^T \mathbf{\Sigma}_k^{-1}(\psi - \mu_k)\right]}{\sqrt{(2\pi)^{n_d}|\mathbf{\Sigma}_k|}}, \tag{11}$$

where $n_c$ is the number of component distributions, $n_d$ is the dimension of the state space, $\tau_k$ is the probability of component distribution $k$, $\mu_k$ and $\mathbf{\Sigma_k}$ are the mean and covariance of the $k$th component distribution. The central idea behind the CEnKF is the addition of a cluster analysis step preceding the analysis scheme of the KF. In this implementation the cluster analysis is responsible for estimating the parameters of the GMM, $\tau_k$, $\mu_k$ and $\mathbf{\Sigma_k}$ from the ensemble of forecast states.

*Expectation maximization.* The EM algorithm produces a fuzzy classification, meaning that a particular state in the forecast ensemble is not necessarily associated with a single Gaussian distribution. Let $n_e$ denote the number of ensemble states and $n_c$ the number of component distributions. Define the $n_e$ by $n_c$ matrix, $w$, $[w_{j,k} = p(\psi_j|\mu_k, \mathbf{\Sigma}_k)/\sum_{l=1}^{n_c} p(\psi_j|\mu_l, \mathbf{\Sigma}_l)]$ this is the probability that the $j$th member of the forecast ensemble is drawn from the $k$th component distribution.

The EM is a two step algorithm with an expectation step and a maximization step. In the expectation step the likelihood of each ensemble member under the component distributions is computed,

$$\mathbf{w} = \left\{ w_{j,k} = \frac{\exp\left[\frac{-1}{2}(\psi_j - \mu_k)^T {\boldsymbol{\Sigma}_k}^{-1}(\psi_j - \mu_k)\right]}{\sqrt{(2\pi)^{n_d}|\boldsymbol{\Sigma}_k|}} \right\} \tag{12}$$

$$w_{j,k} \rightarrow \frac{w_{j,k}}{\sum_{l=1}^{n_c} w_{j,l}}. \tag{13}$$

In the maximization step optimal parameters are chosen for the current weights. For eack class $k$,

$$n_k = \sum_{j=1}^{n_e} w_{j,k} \tag{14}$$

$$\tau_k = \frac{n_k}{n_e} \tag{15}$$

$$\mu_k = \sum_{j=1}^{n_e} w_{j,k}\psi_j / n_k \tag{16}$$

$$\boldsymbol{\Sigma}_k = \sum_{j=1}^{n_e} w_{j,k}(\psi_j - \mu_k)(\psi_j - \mu_k)^T / n_k. \tag{17}$$

To begin the algorithm the covariances are set to the ensemble covariance and the means are randomly selected from the $\psi_j$. The expectation and maximization steps are repeated until convergence of the $w$, $\mu$ and $\boldsymbol{\Sigma}$ are reached. Because the EM algorithm monotonically increases the total data likelihood, $L(\{\psi_j\} \mid \{\mu_k, \boldsymbol{\Sigma}_\mathbf{k}, w_{j,k}\})$, in each iteration it can be shown to converge under fairly mild conditions (Fraley and Raftery, 2002).

In fitting a GMM to data points $\psi_j$, the first step is selecting the number of component distributions, $n_c$. In some problems $n_c$ can be chosen based on prior knowledge of the distribution, for example, in a ensemble integration of the double well model (Miller et al., 1999) it is sensible to choose $n_c = 2$ at all times because the model pdf is known to be bimodal. In settings where $n_c$ can not be determined a priori a computational method is needed to pick $n_c$ based on the forecast ensemble.

One method commonly used to select the number of constituent distributions, is to choose the value of $n_c$ which minimizes Akaie's information criteria (AIC) (Hu and Xu, 2004),

$$AIC(k) = -2\sum_{i=1}^{n_e} \log p(\psi_i|\hat{\theta}_k) + 2D_k \tag{18}$$

$$n_c = \text{argmin(AIC)}. \tag{19}$$

Here $D_k$ is the number of parameters in the GMM with $k$ components and $\hat{\theta}_k$ is the best estimate of the parameters in the GMM with $k$ components. The optimal parameters, $\hat{\theta}_k$, are determined using the EM algorithm. The $AIC(k)$ are an approximation of the integrated likelihood of a $k$ component GMM given the forecast ensemble and a uniform prior over $n_c$(Fraley and Raftery, 2002), that is,

$$-2\text{Log}\prod_{i=1}^{n_e} p(\psi_i^f | n_c = k) \simeq AIC(k). \tag{20}$$

## 2.2. CEnKF algorithm

The Monte Carlo integration in the CEnKF operates in exactly the same way as the EnKF. The CEnKF analysis step first estimates a mixture model for the forecast ensemble using EM. Next, a Kalman gain matrix is computed for each component distribution. The linear KF update is made based on each component distribution's Kalman gain matrix, leading to a weighted ensemble of size $n_e n_c$. Finally, the weighted ensemble is remapped in an $n_c \rightarrow 1$ fashion according to the likelihood of the component distributions given the observed data. Sequentially the analysis step works as follows:

(i) The first portion of the analysis step in the CEnKF is determining the number of component distributions, $n_c$ using Akaie's information criteria (eq. 18). If $n_c = 1$ the CEnKF analysis step reduces to the standard EnKF analysis.

(ii) Apply the EM algorithm to the ensemble of states, $\{\psi_i\}$. This returns, $w$ as well as the estimates of $\tau_k$, $\mu_k$ and $\boldsymbol{\Sigma}_\mathbf{k}$ for each of the component distributions.

(iii) For each component distribution, $k$, compute:

$$\mathbf{P}[k]^f\mathbf{H}^T = \sum_{j=1}^{n_e} w_{j,k}\left(\psi_j^f - \mu_k\right)\left(\mathbf{H}\psi_j^f - \mathbf{H}\mu_k\right)^T / n_k \tag{21}$$

$$\mathbf{HP}[k]^f\mathbf{H}^T = \sum_{j=1}^{n_e} w_{j,k}\left(\mathbf{H}\psi_j^f - \mathbf{H}\mu_k\right)\left(\mathbf{H}\psi_j^f - \mathbf{H}\mu_k\right)^T / n_k \tag{22}$$

and the Kalman gain matrix for the component distribution

$$\mathbf{K}[k] = \mathbf{P}[k]^f\mathbf{H}^T(\mathbf{HP}[k]^f\mathbf{H}^T + \mathbf{R})^{-1}. \tag{23}$$

(iv) Compute the Kalman update for each ensemble member $j$ under each component distribution $k$,

$$\psi_j^{a,k} = \psi_j^f + \mathbf{K}[k]\left(d - \mathbf{H}\psi_j^f - e_j\right) \tag{24}$$

producing an ensemble of size $n_e n_c$.

(v) Calculate the conditional likelihood of each component distribution based on the observed data $d$,

$$\tau_k^a = p(\mu_k, \boldsymbol{\Sigma}_k, \mathbf{R}|d) = \frac{p(d|\mathbf{R}, \mu_k, \boldsymbol{\Sigma}_k)n_k}{\sum_{j=1}^{n_c} p(d|\mathbf{R}, \mu_j, \boldsymbol{\Sigma}_j)n_j}, \tag{25}$$

where

$$p(d|R, \mu_k, \boldsymbol{\Sigma}_k) = \frac{\exp\left[-\frac{1}{2}(d - \mathbf{H}\mu_k)^T(\mathbf{H}\Sigma_\mathbf{k}\mathbf{H}^T + \mathbf{R})^{-1}(d - \mathbf{H}\mu_k)\right]}{\sqrt{(2\pi)^m|(\mathbf{H}\Sigma_\mathbf{k}\mathbf{H}^T + \mathbf{R})|}} \tag{26}$$

and $m$ is the number of observations.

(vi) Define the analysis image of the GMM,

$$\mu_k^a = \sum_{j=1}^{n_e} w_{j,k}\psi_j^{a,k}/n_k, \tag{27}$$

$$\mathbf{P}[k]^a = \sum_{j=1}^{n_e} w_{j,k}\left(\psi_j^{a,k} - \mu_k^a\right)\left(\psi_j^{a,k} - \mu_k^a\right)^T/n_k. \tag{28}$$

In this step I project each family of analysis ensemble state, $\psi_n^{a,k}$ to a standard normal random variable using the weights from the step (ii), $\psi_n^{sn} \rightarrow \sum_{k=1}^{n_c} w_{n,k}{\mathbf{S_k}}^{-1}(\psi_n^{a,k} - \mu_k^a)$. The normalized variables are then mapped back to the component distributions in the GMM with weights depending on the posterior likelihood of each component distribution, $\psi_n^a \rightarrow \sum_{k=1}^{n_c} \tau_k^a(\mu_k^a + \mathbf{S_k}\psi_n^{sn})$. In summary the remapped analysis ensemble is:

$$\psi_n^a = \sum_{j=1}^{n_c} \tau_j^a \left\{ \mu_j^a + \mathbf{S_j}\left[\sum_{k=1}^{n_c} w_{n,k}{\mathbf{S_k}}^{-1}\left(\psi_n^{a,k} - \mu_k^a\right)\right]\right\}. \tag{29}$$

### 2.3. Derivation of the CEnKF algorithm

The CEnKF analysis ensemble is designed to sample a Gaussian approximation to the posterior distribution under the assumption that the prior distribution is a GMM. As in the KF the posterior image of each component distribution is assumed to be linear in the observations.

The distribution we wish to sample is the Gaussian approximation to:

$$p(\psi|d) = \frac{p(d|\psi)p(\psi)}{B} = \sum_{k=1}^{n_c} \tau_k \frac{p(d|\psi)p(\psi|\mu_k, \mathbf{\Sigma}_k)}{B}, \tag{30}$$

where $B$ is the Bayes factor, $B = \int p(d|\psi)p(\psi)d\psi$. Let $B_k = \int p(d|\psi)p(\psi|\mu_k, \mathbf{\Sigma}_k)d\psi$ denote the Bayes factor for the $k$th component distribution. Applying Bayes theorem to the $k$th component distribution,

$$p(\psi|d) = \sum_{k=1}^{n_c} \tau_k \frac{p(\psi|d, \mu_k, \Sigma_\mathbf{k})B_k}{B} \propto \sum_{k=1}^{n_c} \tau_k^a\, p(\psi|d, \mu_k, \Sigma_\mathbf{k}). \tag{31}$$

Now it is established that the posterior pdf is a $n_c$ component mixture model. The probability of the $k$th component distribution is $\tau_k^a$ of step (vi) and the component distributions are the posterior images of the prior component distributions. To sample the component distribution posteriors, $p(\psi|d, \mu_k, \Sigma_\mathbf{k})$ we use the EnKF update on the prior image of the $k$th component distribution. The prior image of the $k$th component distribution is represented by the weighted ensemble $\{\psi_j\}_{j=1}^{n_e}$ with weights, $w_{j,k} = p(\psi_j \mid \mu_k, \mathbf{\Sigma_k})$. The posterior image of the $k$th component distribution is assumed to be the local Kalman update of this weighted ensemble,

$$\{\psi_j^{a,k}\}_{j=1}^{n_e} \tag{32}$$

again with weights $w_{j,k}$.

After computing the weighted ensembles $\psi_j^{a,k}$, I choose to return to an ensemble of size $n_e$ with uniform weights drawn from the single Gaussian distribution with mean

$$\mu^a = \sum_{k=1}^{n_c} \tau_k^a \mu_k^a \tag{33}$$

and covariance

$$\mathbf{\Sigma}^a = \sum_{k=1}^{n_c} \tau_k^a \mathbf{\Sigma}_k^a \tag{34}$$

which is the best Gaussian approximation to the posterior GMM. The advantages of this approach are that ensemble bifurcation is limited at analysis times and a uniformly weighted ensemble is created in the posterior without resampling.[1]

### 2.4. Approximation for high dimensional models

Because steps (i), (ii) and (vi) of the CEnKF algorithm rely on computing, inverting, and finding the Cholsky decomposition of full $n_d$ by $n_d$ matrices the CEnKF algorithm is not practical for high dimensional models as presented. Here an approximation to the CEnKF algorithm is introduced which assumes the model covariances, $\mathbf{\Sigma_k}$ and $\mathbf{P^a}[k]$, are diagonal in steps (i), (ii) and step (vi), respectively. In steps (iii) through (v) the full covariance is used in computing the Kalman gains. This approximation is appropriate for large-scale applications, and requires no more storage than $n_c$ times the storage requirements of the EnKF. Throughout the rest of the the paper this approximation will be referred to as the $C_D$EnKF algorithm ($D$ for diagonal).

## 3. Steele–Henderson *P–Z* model

As a representative non-linear stochastic model I use the Steele–Henderson phytoplankton–zooplankton model (SH92). This is the 2-D model with state space $\psi = (P, Z)$ and dynamics given by the stochastic differential equation,

$$\mathrm{d}P = \beta P\left(1 - \frac{P}{\gamma}\right) - \frac{\lambda P^\nu}{\mu^\nu + P^\nu} Z\,\mathrm{d}t \tag{35}$$

$$\mathrm{d}Z = \alpha\frac{\lambda P^\nu}{\mu^\nu + P^\nu} Z - \alpha\delta Z^m\,\mathrm{d}t + \sigma_Z\xi. \tag{36}$$

I use the normalized equations, $\alpha = \beta = \lambda = \mu = 1$, $\gamma = 10$ and $\delta = \frac{3}{4}.\gamma$ is the carrying capacity for the phytoplankton population. $\mu$ is the half saturation population. I use the linear closure with $\nu = 1$ and $m = 1$.

[1] Another reasonable choice for step (vi) would be to sample an analysis ensemble from the weighted ensembles (eq. 32), avoiding the approximation of the posterior distribution with a Gaussian distribution. This could be accomplished by standard importance resampling as described in van Leeuwen (2003). The testing of the algorithm with resampling is beyond the scope of this paper.

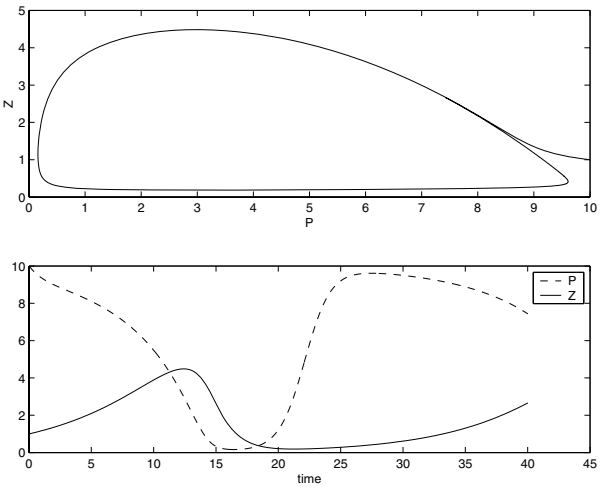


*Fig. 1.* Trajectory of the SH92 model without stochastic forcing term $+\sigma_Z \xi_1$. The upper frame shows the state-space $P$ versus $Z$ trajectory. The lower frame shows the time series of the two populations.

Here the model is stochastically forced in the zooplankton term. The $\xi$ is a Wiener process (i.e. one-dimensional diffusion) and $\sigma_Z = \frac{1}{10}$. The initial populations are assumed to be known perfectly, $P = 10$ and $Z = 1$. The unforced model exhibits a classical predator-prey limit cycle (Fig. 1). With the stochastic forcing of the zooplankton population the model takes significant departures from the unforced trajectory, as the point of time in which predation overcomes growth for the zooplankton becomes random. In the forced model the cycle is still discernible (Fig. 2).

The system of stochastic equations is solved with Eulerian time stepping with $\Delta t = 0.02$ d. The measurement operator records only the phytoplankton variable, $P$, every 500 model time steps (10 d). The measurement has a standard error of 0.01. This model is chosen for the way in which the covariance between the phytoplankton and zooplankton populations changes at different points in state space. The highly non-Gaussian pdf over state space at 10 d (Fig. 3) exhibits two types of covariance behaviour. If the phytoplankton population is high ($P > 3$) then the zooplankton population is negatively correlated with the phytoplankton. If the phytoplankton population is low ($P < 3$) they are positively correlated. Overall the correlation coefficient between P and Z is $\sigma_{PZ} = -0.42$, in the region where $P > 3\,\sigma_{PZ} = -0.94$, and in the region where $P < 3\,\sigma_{PZ} = 0.74$. A GMM is capable of approximating this distribution because within each region ($P < 3$, $P > 3$) the distribution is approximately Gaussian. Of course the biological model has strictly positive populations so the negative tails of the component distributions are inconsistent with the model, however this error is less important for the component distributions in the GMM which have smaller variances than the overall Gaussian approximation to the distribution.

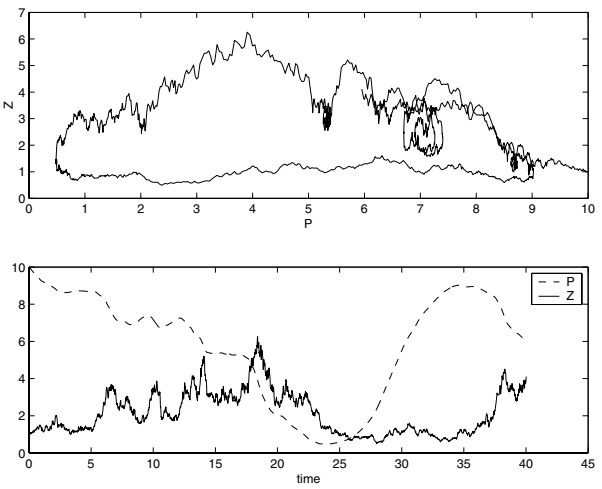


*Fig. 2.* Same as Fig. 1 except with stochastic forcing of the zooplankton population.

### 3.1. Twin experiments

Twin experiments were carried out to test the CEnKF and $C_D$EnKF accuracy versus the EnKF accuracy. Five hundred experiments were conducted. In each experiment a 'true' state is created by simulating the stochastically forced SH92 model (eqs 35–36). Ensembles of size 100 were used for each of the filters and the analysis states were compared with the same 'true' state. A maximum of $n_c = 4$ component distributions is imposed in the AIC search. The CEnKF ensemble both narrows the range of the zooplankton prediction and also the ensemble mean of the CEnKF makes a more accurate prediction of the truth (Figs. 4 and 5, Table 2). This in turn leads to a more accurate forecast of the zooplankton population (Fig. 6). I test the ensemble statistics by checking the frequency with which the prediction (ensemble mean) is more than one ensemble standard deviation from the truth (Fig. 7). Overall both the EnKF, CEnKF and $C_D$EnKF give reasonable one standard deviation confidence intervals, given the non-linear dynamics at play. The CEnKF produces confidence intervals no worse than the EnKF (Fig. 7, Table 2).

To assess the effect of non-Gaussian contributions to the performance of the EnKF and CEnKF a set of experiments were conducted with increasing time between observations. At short times the ensemble of particles will not spread far enough to exhibit significant non-Gaussian structure, such as multimodal behaviour or covariance which changes structure across the ensemble. At longer times the ensemble spreads to form a highly non-Gaussian pdf (e.g. Fig. 3). As has been demonstrated before, (Miller et al., 1999; Evensen, 2006), the EnKF performs well if the time between the observations is short and the ensemble does not become excessively non-Gaussian. The time between observations in the experiment runs from $t = 1, 2, \ldots, 10$. For each sampling interval 50 twin experiments are conducted to

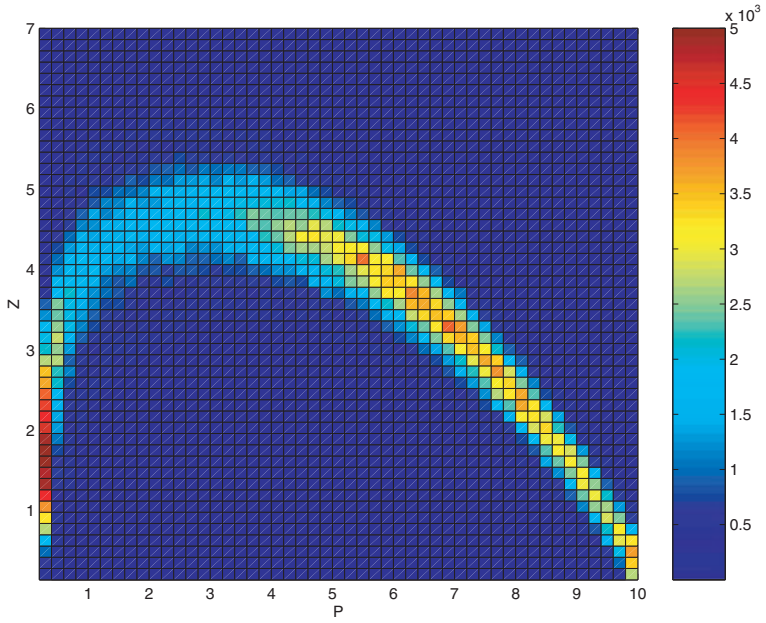


*Fig. 3.* Two-dimensional histogram of the SH92 model after 10 d with initial condition $P = 10$ and $Z = 1$. An ensemble of size $10^5$ is used to approximate the pdf. The ensemble is generated by simulating the stochastic differential, equations 35 and 36, independently. Each state is run forward to $t = 500$ d with a time step of $\Delta t = 0.02$ d. The colour scale is the probability that the model is in a particular bin at the end of the simulation.

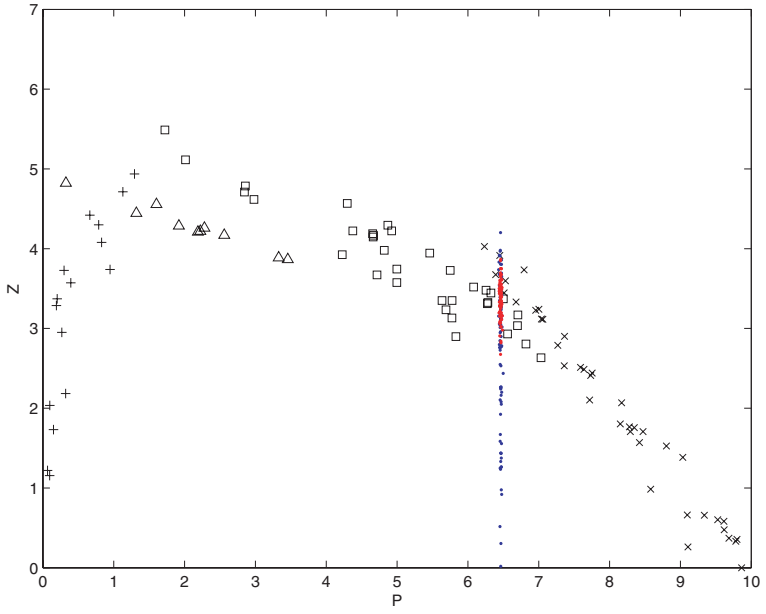


*Fig. 4.* Example of the analysis step in the CEnKF algorithm. Each ensemble member is plotted with a square, +, △, or *x* depending on which of the four component distributions it is most associated with. The blue dots are the EnKF analysis ensemble and the red dots are the CEnKF analysis ensemble based on the full forecast ensemble.

determine the mean behaviour of the filters. As expected the error statistics for the EnKF and CEnKF agree at short sampling intervals (Fig. 8). However, as the time between observations becomes larger (evident for sampling intervals greater than 5 d) the EnKF is no longer able to accurately estimate the zooplankton population, and so the forecast skill suffers relative to the CEnKF and $C_D$EnKF filters.

For the SH92 model the difference in the forecast error between the CEnKF and $C_D$EnKF filters is negligible (Fig. 8). However the $C_D$EnKF algorithm uses more clusters than the CEnKF, especially at short sampling intervals. This is due to the approximation of tilted ensembles by a set of covariances oriented with the axes, rather than a single ellipse tilted to match the ensemble orientation.

## 4. Considerations and conclusions

I have demonstrated that the CEnKF can achieve both higher accuracy and precision than the traditional EnKF when applied to the SH92 *P–Z* model. This can be achieved while maintaining the accuracy of the ensemble forecast statistics. These conclusions are dependent on the partial measurement operators used here,

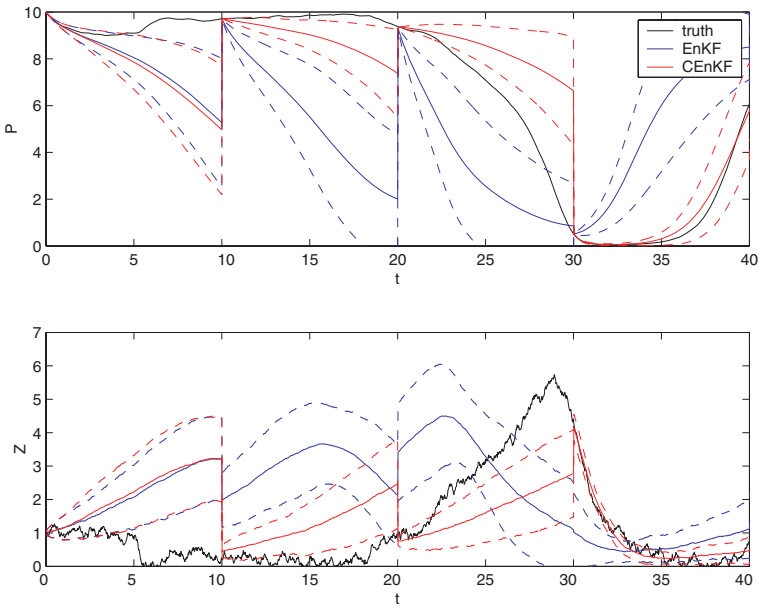


*Fig. 5.* Time series plot from the experiment depicted in Fig. 6. The upper frame is the phytoplankton time series. The lower frame is the time series of zooplankton population. The solid blue line is the ensemble mean of the EnKF ensemble. The dashed blue lines are the ensemble mean ± the ensemble standard deviation. The red lines are the same for the CEnKF ensemble. The black lines represent the truth. The EnKF and CEnKF analysis are carried out at $t = 10$, 20, 30 and 40.

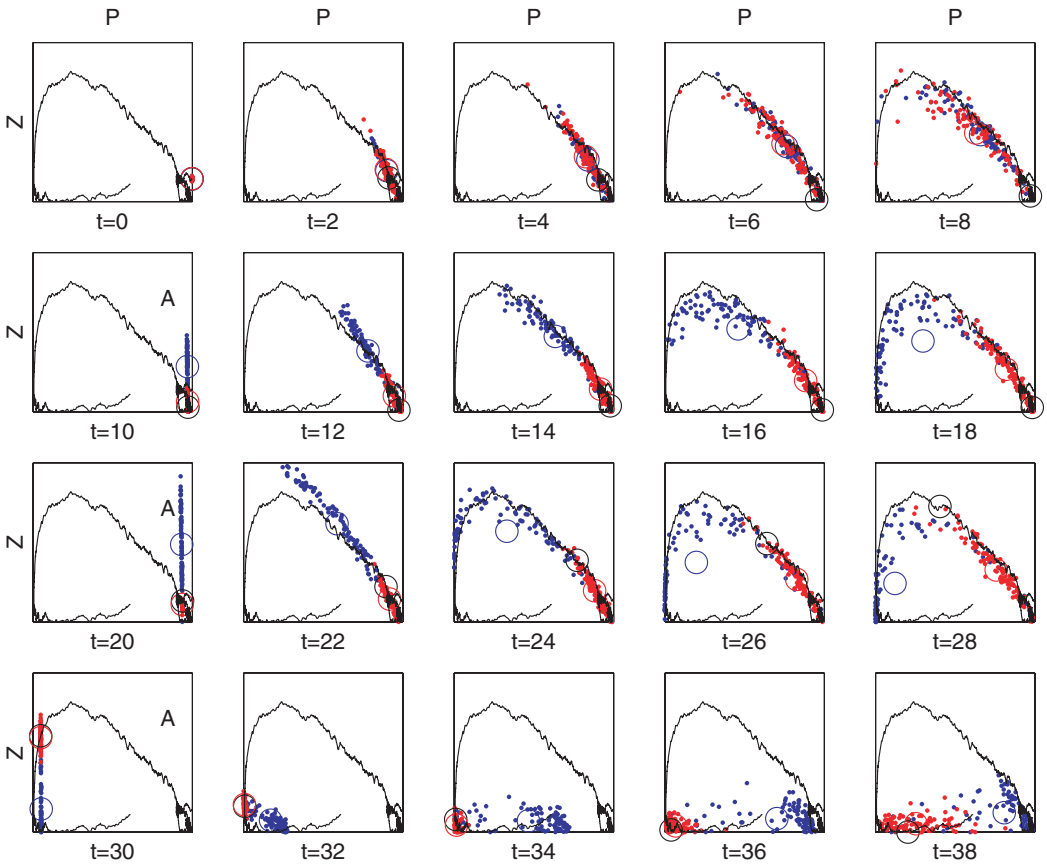


*Fig. 6.* Ensemble of states for one twin experiment. The blue dots are the states in the EnKF ensemble, the red dots are the states in the CEnKF ensemble. The red circles are the CEnKF ensemble mean and the blue circles are the EnKF ensemble mean. The black circles denotes the true value. The black line shows the state space trajectory of the true state. The phytoplankton axis runs from 0 to 10 and the zooplankton axis runs from 0 to 7. The ensemble correction occurs at $t = 10$, 20 and 30, at which time I show the analysis ensemble. Because the measurement of the phytoplankton is accurate relative to the forecast variance, at update times the ensembles from both methods appear like vertical lines.

*Table 2.* Summary of the results for the EnKF, CEnKF and $C_D$EnKF in the twin experiments with the SH92 model. All time averages are computed for the period after the first observation is made

| Statistic | *EnKF* | *CEnKF* | *CEnKFD* |
|---|---|---|---|
| $\sqrt{E[(\overline{P}-P^t)^2]}$ | 1.81 | 1.19 | 1.22 |
| $\sqrt{E[(\overline{Z}-Z^t)^2]}$ | 1.05 | 0.79 | 0.80 |
| $\sqrt{\overline{(P-\overline{P})^2}}$ | 1.46 | 0.79 | 0.88 |
| $\sqrt{\overline{(Z-\overline{Z})^2}}$ | 0.82 | 0.53 | 0.60 |
| $(\overline{P}-P^t)^2 > \overline{(P-\overline{P})^2}$ | 0.74 | 0.65 | 0.65 |
| $(\overline{Z}-Z^t)^2 > \overline{(Z-\overline{Z})^2}$ | 0.68 | 0.63 | 0.64 |

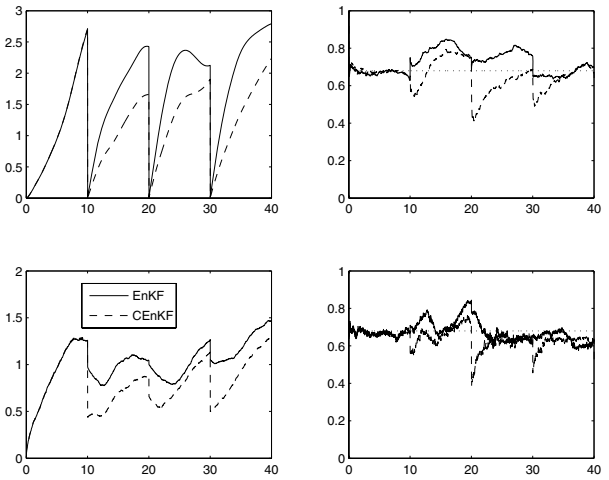


*Fig 7.* The left column is the average error as a function of time for $P$ (upper left frame) and $Z$ (lower left frame). The EnKF ensemble is shown with the solid line, the CEnKF with the dashed line. The right column is the probability that the ensemble average is within one ensemble standard deviation of the truth. If the error distribution is Gaussian and correct, this should be about 0.68, the dotted line.

as well as the length of the forecasts between measurements. In tests with shorter forecasts the AIC usually selected a one or two component GMM, leading to results equivalent to the EnKF.

Though not examined in the experiments presented here, the ensemble size required for the CEnKF analysis should be closely related to the ensemble size needed for the EnKF. It is reasonable to expect that if the $n_k$ (eq. 14) are all as large as the requisite ensemble size for the EnKF, then an adequate representation of the local means and covariances is present to estimate the Kalman gains, $\mathbf{K}[k]$. In the application tested here the $n_k$ were rarely less than $\frac{1}{4}n_e$ when $n_c = 2$, hence one would expect that no more than an ensemble four times larger than required for the EnKF would be necessary for the CEnKF. The procedure for selecting $n_c$, step (i), could also be amended to guarantee adequately large $n_k$ to estimate the mean and covariance of each constituent distribution.

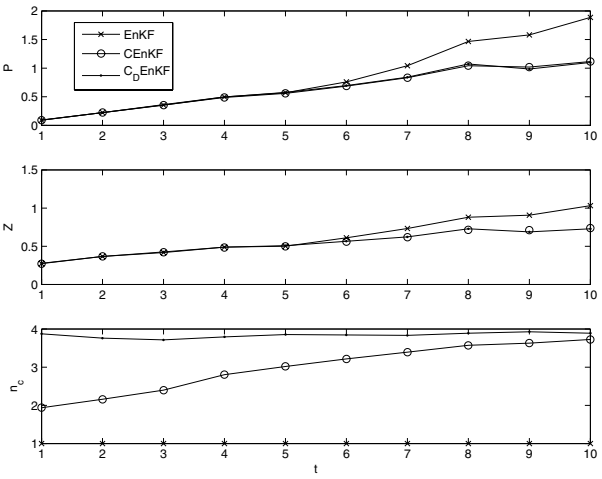


*Fig 8.* The top frame is the rms error of $P$ as a function of time between observations, the middle frame is the same for the $Z$ variable. The bottom frame is the average number of component distributions selected for each method.

In addition to the application to the SH92 model, the CEnKF was also tested with the stochastic Lorenz equations (Lorenz, 1963; Miller et al., 1999). The performance of the CEnKF relative to the EnKF was quite similar to the results presented here. In higher dimensional models, such as realistic numerical weather or ocean circulation models, the state space has a dimension much higher than the two dimensions of the model examined here. It is generally assumed that $n_d$ is large enough that one will need to avoid inverting full $n_d$ by $n_d$ matrices for such models. This imposes restrictions on the form of covariance used in the cluster analysis step (steps i and ii). One option is the restriction to a diagonal covariance matrix, implemented in the $C_D$EnKF algorithm. No decrease in accuracy occurred when using the diagonal covariance matrix in the cluster analysis step. Alternatively, the state space could be projected onto a lower dimensional space depicting some relevant phenomenon, and the full covariance matrix in this state space could be used.

Though the EM algorithm guarantees convergence to a local minimum of the likelihood function under general conditions, the rate of convergence is problem dependent (Fraley and Raftery, 2002). In addition to the number of parameters being estimated and the number of data points ($n_e$), the convergence speed depends on how well separated the ensemble states are. The EM algorithm is amenable to parallelization, so application of the algorithm to high dimensional models should not be problematic, provided the covariances are approximated by diagonal matrices, as in the $C_D$EnKF, with $n_d$ parameters, rather than full covariance matrices with $n_d(n_d+1)/2$ parameters.

Lastly the remapping step (step vi) will need to be simplified in high dimensional problems, as the remapping requires Cholsky decomposition of the covariance matrix and subsequent inversion of full $n_d$ by $n_d$ matrices. One could use a simple resampling

of the weighted posterior ensemble (eq. 32), instead of the mapping used here. This approach also avoids the approximation of the posterior distribution with a Gaussian distribution. Approximation of the covariance matrix with a diagonal matrix can also be used in the remapping step as well. This approach was used in the $C_D$EnKF algorithm and shown to be as effective as the remapping based on the full covariance matrix and is entirely suitable for high dimensional models.

The implementation and testing of this procedure with realistic numerical models and real data is left for future work.

## 5. Acknowledgments

This work was supported by NSF grant DMS-0417845.

## References

Eknes, M. and Evensen, G. 2002. An ensemble Kalman filter with a 1-D marine ecosystem model. *J. Marine Sys.* **36**, 75–101.

Evensen, G. 1994. Sequential data assimilation with a non-linear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *J. Geophys. Res.* **97**(C11), 17 905–17 924.

Evensen, G. 2003. The ensemble Kalman filter: theoretical formulation and practical implementation. *Ocean Dyn.* **53**, 343–367.

Evensen, G. 2006. *Data Assimilation: The Ensemble Kalman Filter*. Springer-Verlag, Berlin, Heidelberg, 279 pp.

Fraley, C. and Raftery, A. 2002. Model based clustering, discriminant analysis, and density estimation. *J. Am. Stat. Assoc.* **97**(458), 611–631.

Hu, X. and Xu, L. 2004. Investigation on several model selection criteria for determining the number of cluster. *Neural Infor. Proc.–Lett. Rev.* **4**(1), 1–10.

Lorenz, E. N. 1963. Deterministic nonperiodic flow. *J. Atmos Sci.* **20**, 448–464.

Miller, R. N., Carter, E. F. and Blue, S. T. 1999. Data Assimilation into non-linear stochastic models. *Tellus* **51A**, 167–194.

Steele, J. H. and Henderson, E. W. 1992. The role of predation in plankton models. *J. Plankton Res.* **14**(1), 157–172.

van Leeuwen, P. J. 2003. A variance minimizing filter for large-scale applications. *Mon. Wea. Rev.* **131**, 2071–2084.