

Kalman Filtering: Part II

Instructor: Istvan Szunyogh

Class: March 2, 2007

The Quest for an Optimal Formulation of the Ensemble Based Kalman Filter

A Summary of Important Results from Part I

- Data assimilation schemes for nonlinear problems are based on minimizing a cost function similar to

$$J_{t_n}^o(\mathbf{x}) = [\mathbf{x} - \bar{\mathbf{x}}_n^b]^T (\mathbf{P}_n^b)^{-1} [\mathbf{x} - \bar{\mathbf{x}}_n^b] + [\mathbf{y}_n^o - \mathbf{H}_n \mathbf{x}]^T \mathbf{R}_n^{-1} [\mathbf{y}_n^o - \mathbf{H}_n \mathbf{x}] + c. \quad (1)$$

- At a minimum, a nonlinear model forces a change in the forecast equations

$$\bar{\mathbf{x}}_n^b = \mathbf{M}_{t_{n-1}, t_n} \bar{\mathbf{x}}_{n-1}^a, \quad (2)$$

$$\mathbf{P}_n^b = \mathbf{M}_{t_{n-1}, t_n} \mathbf{P}_{n-1}^a \mathbf{M}_{t_{n-1}, t_n}^T. \quad (3)$$

while nonlinear observation operators \mathbf{H}_n force a change in the analysis equations (4) and (5)

$$\bar{\mathbf{x}}_n^a = \bar{\mathbf{x}}_n^b + \mathbf{P}_n^a \mathbf{H}_n^T \mathbf{R}_n^{-1} (\mathbf{y}_n^o - \mathbf{H}_n \bar{\mathbf{x}}_n^b) = \bar{\mathbf{x}}_n^b + \mathbf{K} (\mathbf{y}_n^o - \mathbf{H}_n \bar{\mathbf{x}}_n^b) \quad (4)$$

$$\mathbf{P}_n^a = (\mathbf{I} + \mathbf{P}_n^b \mathbf{H}_n^T \mathbf{R}_n^{-1} \mathbf{H}_n)^{-1} \mathbf{P}_n^b = (\mathbf{I} - \mathbf{K} \mathbf{H}) \mathbf{P}_n^b. \quad (5)$$

Summary of Part I: The Extended Kalman Filter

- Computes $\bar{\mathbf{x}}_n^b = M_{t_{n-1}, t_n}(\bar{\mathbf{x}}_{n-1}^a)$ using the nonlinear model
- Computes \mathbf{P}_n^b using the linearization $\mathbf{M}_{t_{n-1}, t_n}$ of M_{t_{n-1}, t_n} around $\bar{\mathbf{x}}_{n-1}^a$.
- The analysis then uses the linearization \mathbf{H}_n of H_n around $\bar{\mathbf{x}}_n^b$.
- For the case when asynchronous observations are assimilated in batches (4-d assimilation), $\bar{\mathbf{x}}_n^b$ is a trajectory (maps $\bar{\mathbf{x}}_{n-1}^a$ to the state at the time of the observations) and f_{H_n} is linearized around the state at the time of the observation

Summary of Part I: Ensemble Kalman Filtering:

The key idea of ensemble Kalman filtering is

- To choose at time t_{n-1} an ensemble of initial conditions, $\{\mathbf{x}_{n-1}^{a(i)} : i = 1, 2, \dots, k\}$, whose spread around $\bar{\mathbf{x}}_{n-1}^a$ characterizes the analysis covariance \mathbf{P}_{n-1}^a
- To propagate each ensemble member using the nonlinear model, and compute $\bar{\mathbf{x}}^b$ and \mathbf{P}_n^b based on the resulting ensemble $\{\mathbf{x}_n^{b(i)} : i = 1, 2, \dots, k\}$ at time t_n .

$$\bar{\mathbf{x}}^b = k^{-1} \sum_{i=1}^k \mathbf{x}^{b(i)},$$

$$\mathbf{P}^b = (k - 1)^{-1} \sum_{i=1}^k (\mathbf{x}^{b(i)} - \bar{\mathbf{x}}^b)(\mathbf{x}^{b(i)} - \bar{\mathbf{x}}^b)^T \quad (6)$$

Remarks:

- The main advantage of the ensemble approach is that it is much cheaper than the direct evaluation of

$$\mathbf{P}_n^b = \mathbf{M}_{t_{n-1}, t_n} \mathbf{P}_{n-1}^a \mathbf{M}_{t_{n-1}, t_n}^T$$

when $k \ll m$. Only the computation of $\mathbf{M}_{t_{n-1}, t_n}$ would require m integrations of the model. Recall that m for a typical NWP model is $10^6 - 10^7$!

- The ensemble approach eliminates the need for an explicit linearization of M
- We have not addressed the issue of nonlinear H , yet. As we will see, in an ensemble based Kalman filter H does not have to be linearized either.
- In the 4-dimensional case, \mathbf{P}_n^b is represented by an ensemble of trajectories

The Challenge

- The rank of \mathbf{P}_n^b can be as large as $m - 1$, while the rank of its estimate based on a k member ensemble cannot be larger than k .
- The practical approach to meet this challenge is *covariance localization*: Covariances are considered only between nearby locations
- All successful implementation of the ensemble based approach use some form of covariance localization.

Building an Ensemble Based Scheme

One can choose between a number of options for the different components of the scheme

Possible Cycling Strategies

- *Cycling by observations:* In each cycle, a group of observations is selected and the state estimate is updated at all grid points at which the state estimate may be affected by the selected observations. In the extreme cases, the observations are assimilated
 - one by one (serial Ensemble Kalman Filter Schemes)
 - all at once (variational methods, that include 3D-Var, 4D-Var and NLMLE of Zupanski 2005)
- *Cycling by state vector components:* In each cycle, first a group of state vector components is selected for update, then all observations that may affect the estimate of the selected state vector components are assimilated in one batch. In the extreme cases, the state vector components are estimated
 - one by one or grid point by grid point (LEKF, LETKF)
 - all at once (variational methods)

Justification of the Cycling Strategies II

These strategies are based on the following observations:

- Cycling the observations: Assume that the observations can be divided up into sets in such a way, that the errors between any two observations from two different sets are uncorrelated. After the assimilation of the observations from the first n sets, the most likely trajectory is the one that minimizes the “cost function”

$$J^o(\{\mathbf{x}(t)\}) = \sum_{j=1}^n [\mathbf{y}_j^o - H_j(\mathbf{x}(t_j))]^T \mathbf{R}_j^{-1} [\mathbf{y}_j^o - H_j(\mathbf{x}(t_j))].$$

This cost function depends only on which observations are assimilated, but it does not depend on the order in which these observations are assimilated

- Cycling by state vector component: The different state vector components are estimated independently in all alternative forms of the update equations

Localization Strategies I

- *Explicit localization*
 - Observations and observation increments are used only from a local region.
 - Typically used when the state estimate is updated grid point by grid point.
 - Observations further away from the grid point may be taken into account with smaller weights.
 - The state estimation is done for a local region. There is no guarantee that it solves the global optimization problem.
 - Easy to implement on a parallel computer

Localization Strategies II

- *Implicit Localization*
 - The background error covariance is tapered to zero moving away from the observational location.
 - Typically serial schemes use this approach.
 - The taper function can have many different shapes.
 - Formally, the state estimation is done globally, but the effect of the observations on the state estimate is local.
 - Parallel implementation may be difficult, though it is not impossible.

Possible Strategies to Obtain the Analysis Ensemble

In an ensemble based Kalman filter the analysis step must

- Determine the state estimate $\bar{\mathbf{x}}^a$
- Generate an ensemble $\{\mathbf{x}^{a(i)} : i = 1, 2, \dots, k\}$ with the appropriate sample mean and covariance:

$$\bar{\mathbf{x}}^a = k^{-1} \sum_{i=1}^k \mathbf{x}^{a(i)},$$

$$\mathbf{P}^a = (k-1)^{-1} \sum_{i=1}^k (\mathbf{x}^{a(i)} - \bar{\mathbf{x}}^a)(\mathbf{x}^{a(i)} - \bar{\mathbf{x}}^a)^T = (k-1)^{-1} \mathbf{X}^a (\mathbf{X}^a)^T,$$

where \mathbf{X}^a is the $m \times k$ matrix whose i th column is $\mathbf{x}^{a(i)} - \bar{\mathbf{x}}^a$. (7)

Possible Strategies to Obtain the Analysis Ensemble

- “Perturbed Observations” method, often called Ensemble Kalman Filter (EnKF) or probabilistic schemes
 - Equation (7) is satisfied only in the limit of k goes to infinity
 - Typically used with serial processing of the observations, though in principle it could be used in a scheme that estimates the state vector components independently
- Square Root Filters, also often called deterministic schemes
 - Equation (7) is satisfied exactly independently of k
 - Both serial and “grid point by grid point” schemes exist

Covariance Inflation

- In practice, an ensemble Kalman filter may fail to synchronize with the “true” system trajectory due to
 - effects of model errors
 - sampling errors due to limited sample size
 - effects of nonlinearities
- Regardless of the cause, underestimating the uncertainty leads to overconfidence in the background state estimate
- This effect is typically compensated by an ad hoc “variance inflation” procedure
 - Hybrid approach—the background error covariance is inflated by the 3D-VAR background error covariance
 - multiplicative inflation of ensemble perturbations
 - additive inflation of ensemble perturbations

LETKF: A local ensemble transform Kalman filter

With the derivation of this scheme the goal was to find the combination of strategies for the individual steps that provides the best overall computational efficiency.

An efficient means of performing the analysis that transforms a background ensemble $\{\mathbf{x}^{b(i)} : i = 1, 2, \dots, k\}$ into an appropriate analysis ensemble $\{\mathbf{x}^{a(i)} : i = 1, 2, \dots, k\}$, using the notation defined above

- Assume that the number of ensemble members k is smaller than both the number of model variables m and the number of observations ℓ , even when localization has reduced the effective values of m and ℓ considerably compared to a global analysis.
- Assume the choice of observations to use for the local analysis to have been performed already, and consider \mathbf{y}^o , H and \mathbf{R} to be truncated to these observations; as such, correlations between errors in the chosen observations and errors in other observations are ignored.

Derivation of the LETKF

- Formally, we want the analysis mean $\bar{\mathbf{x}}^a$ to minimize the Kalman filter cost function (1), modified to allow for a nonlinear observation operator H :

$$J(\mathbf{x}) = (\mathbf{x} - \bar{\mathbf{x}}^b)^T (\mathbf{P}^b)^{-1} (\mathbf{x} - \bar{\mathbf{x}}^b) + [\mathbf{y}^o - H(\mathbf{x})]^T \mathbf{R}^{-1} [\mathbf{y}^o - H(\mathbf{x})]. \quad (8)$$

- However, the $m \times m$ background covariance matrix $\mathbf{P}^b = (k-1)^{-1} \mathbf{X}^b (\mathbf{X}^b)^T$ can have rank at most $k-1$, and is therefore not invertible
- Nonetheless, its inverse is well-defined on the space S spanned by the background ensemble perturbations, that is, the columns of \mathbf{X}^b . Thus J is also well-defined for $\mathbf{x} - \bar{\mathbf{x}}^b$ in S , and the minimization can be carried out in this subspace.

Choosing the Coordinate System I

- In order to perform the analysis on S , we must choose an appropriate coordinate system.
- A natural approach is to use the singular vectors of \mathbf{X}^b (the eigenvectors of \mathbf{P}^b) to form a basis for S (Anderson 2001; Ott et al. 2004). This leads to the Local Ensemble Kalman Filter (LEKF) scheme, the predecessor of the LETKF.
- In the LETKF, we avoid this step by using instead the columns of \mathbf{X}^b to span S , as in Bishop et al. (2004)
- One conceptual difficulty in this approach is that the sum of these columns is zero, since they are defined by the departure from the mean $\bar{\mathbf{x}}^b$. Thus, they are necessarily not linearly independent (i.e., they cannot be considered to be a basis). We could assume the first $k-1$ columns to be independent and use them as a basis, but this assumption is unnecessary and clutters the resulting equations.

Choosing the Coordinate System II

- Instead, we regard \mathbf{X}^b as a linear transformation from a k -dimensional space \tilde{S} onto S , and perform the analysis in \tilde{S} . Let \mathbf{w} denote a vector in \tilde{S} ; then $\mathbf{X}^b\mathbf{w}$ belongs to the space S spanned by the background ensemble perturbations, and $\mathbf{x} = \bar{\mathbf{x}}^b + \mathbf{X}^b\mathbf{w}$ is the corresponding model state.
- Notice that if \mathbf{w} is a Gaussian random vector with mean $\mathbf{0}$ and covariance $(k-1)^{-1}\mathbf{I}$, then $\mathbf{x} = \bar{\mathbf{x}}^b + \mathbf{X}^b\mathbf{w}$ is Gaussian with mean $\bar{\mathbf{x}}^b$ and covariance $\mathbf{P}^b = (k-1)^{-1}\mathbf{X}^b(\mathbf{X}^b)^T$.

- This motivates the cost function

$$\tilde{J}(\mathbf{w}) = (k-1)\mathbf{w}^T\mathbf{w} + [\mathbf{y}^o - H(\bar{\mathbf{x}}^b + \mathbf{X}^b\mathbf{w})]^T \mathbf{R}^{-1} [\mathbf{y}^o - H(\bar{\mathbf{x}}^b + \mathbf{X}^b\mathbf{w})] \quad (9)$$

on \tilde{S}

- In particular, it is claimed that if $\bar{\mathbf{w}}^a$ minimizes \tilde{J} , then $\bar{\mathbf{x}}^a = \bar{\mathbf{x}}^b + \mathbf{X}^b\bar{\mathbf{w}}^a$ minimizes the cost function J .

Choosing the Coordinate System III

- Substituting the change of variables formula into (8) and using (6) yields the identity

$$\tilde{J}(\mathbf{w}) = (k-1)\mathbf{w}^T(\mathbf{I} - (\mathbf{X}^b)^T[\mathbf{X}^b(\mathbf{X}^b)^T]^{-1}\mathbf{X}^b)\mathbf{w} + J(\bar{\mathbf{x}}^b + \mathbf{X}^b\mathbf{w}). \quad (10)$$

- The matrix $\mathbf{I} - (\mathbf{X}^b)^T[\mathbf{X}^b(\mathbf{X}^b)^T]^{-1}\mathbf{X}^b$ is the orthogonal projection onto the null space N of \mathbf{X}^b . (Generally N will be one-dimensional, spanned by the vector $(1, 1, \dots, 1)^T$, but it could be higher-dimensional.)
- Thus, the first term on the right side of (10) depends only on the component of \mathbf{w} in N , while the second term depends only on its component in the space orthogonal to N (which is in one-to-one correspondence with S under \mathbf{X}^b). Thus if $\bar{\mathbf{w}}^a$ minimizes \tilde{J} , then it must be orthogonal to N , and the corresponding vector $\bar{\mathbf{x}}^a$ minimizes J .

Nonlinear Observation Operator

- The most accurate way to allow for a nonlinear observation operator H would be to numerically minimize \tilde{J} in the k -dimensional space \tilde{S} , as in Zupanski (2005).
 - If H is sufficiently nonlinear, then \tilde{J} could have multiple minima, but a numerical minimization using $\mathbf{w} = \mathbf{0}$ (corresponding to $\mathbf{x} = \bar{\mathbf{x}}^b$) as an initial guess would still be a reasonable approach.
 - Having determined $\bar{\mathbf{w}}^a$ in this manner, one would compute the analysis covariance $\tilde{\mathbf{P}}^a$ in \tilde{S} from the second partial derivatives of \tilde{J} at $\bar{\mathbf{w}}^a$
 - Then use \mathbf{X}^b to transform the analysis results into the model space, as below.
- But in order to formulate the analysis more explicitly, we now linearize H about the background ensemble mean $\bar{\mathbf{x}}^b$. Of course, if H is linear then we will find the minimum of \tilde{J} exactly.

Linearizing the Observation Operator in the Ensemble Space

- Since we only need to evaluate H in the ensemble space (or equivalently to evaluate $H(\bar{\mathbf{x}}^b + \mathbf{X}^b \mathbf{w})$ for \mathbf{w} in \tilde{S}), the simplest way to linearize H is to apply it to each of the ensemble members $\mathbf{x}^{b(i)}$ and interpolate.
- To this end, we define an ensemble $\mathbf{y}^{b(i)}$ of background observation vectors by

$$\mathbf{y}^{b(i)} = H(\mathbf{x}^{b(i)}). \quad (11)$$

- We define also their mean $\bar{\mathbf{y}}^b$, and the $\ell \times k$ matrix \mathbf{Y}^b whose i th column is $\mathbf{y}^{b(i)} - \bar{\mathbf{y}}^b$.
- We then make the linear approximation

$$H(\bar{\mathbf{x}}^b + \mathbf{X}^b \mathbf{w}) \approx \bar{\mathbf{y}}^b + \mathbf{Y}^b \mathbf{w}. \quad (12)$$

The same approximation is used in, for example, Houtekamer and Mitchell (2001), and is equivalent to the joint state-observation space method in (Anderson 2001).

Analysis

- This approximation yields the quadratic cost function

$$\tilde{J}^*(\mathbf{w}) = (k - 1)\mathbf{w}^T \mathbf{w} + [\mathbf{y}^o - \bar{\mathbf{y}}^b - \mathbf{Y}^b \mathbf{w}]^T \mathbf{R}^{-1} [\mathbf{y}^o - \bar{\mathbf{y}}^b - \mathbf{Y}^b \mathbf{w}]. \quad (13)$$
- This cost function is in the form of the Kalman filter cost function, using the background mean $\bar{\mathbf{w}}^b = \mathbf{0}$ and background covariance matrix $\tilde{\mathbf{P}}^b = (k - 1)^{-1} \mathbf{I}$, with \mathbf{Y}^b playing the role of the observation operator.

- The analogues of the analysis equations (4) and (5)

$$\bar{\mathbf{x}}_n^a = \bar{\mathbf{x}}_n^b + \mathbf{P}_n^a \mathbf{H}_n^T \mathbf{R}_n^{-1} (\mathbf{y}_n^o - \mathbf{H}_n \bar{\mathbf{x}}_n^b)$$

$$\mathbf{P}_n^a = \left[(\mathbf{P}_n^b)^{-1} + \mathbf{H}_n^T \mathbf{R}_n^{-1} \mathbf{H}_n \right]^{-1}.$$

are then

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

$$\tilde{\mathbf{P}}^a = [(k - 1)\mathbf{I} + (\mathbf{Y}^b)^T \mathbf{R}^{-1} \mathbf{Y}^b]^{-1}. \quad (15)$$

- In model space, the analysis mean and covariance are then

$$\bar{\mathbf{x}}^a = \bar{\mathbf{x}}^b + \mathbf{X}^b \bar{\mathbf{w}}^a, \quad (16)$$

$$\mathbf{P}^a = \mathbf{X}^b \tilde{\mathbf{P}}^a (\mathbf{X}^b)^T. \quad (17)$$

Analysis Ensemble

- In order to initiate the ensemble forecast that will produce the background for the next analysis, we must choose an analysis ensemble whose sample mean and covariance are equal to $\bar{\mathbf{x}}^a$ and \mathbf{P}^a . This amounts to choosing a matrix \mathbf{X}^a so that
 - the sum of its columns is zero
 - $\mathbf{P}^a = (k - 1)^{-1}\mathbf{X}^a(\mathbf{X}^a)^T$ holds.
- The analysis ensemble is formed by adding $\bar{\mathbf{x}}^a$ to each of the columns of \mathbf{X}^a .
- Our choice of analysis ensemble is described by $\mathbf{X}^a = \mathbf{X}^b\mathbf{W}^a$, where

$$\mathbf{W}^a = [(k - 1)\tilde{\mathbf{P}}^a]^{1/2} \quad (18)$$

and by the 1/2 power of a symmetric matrix we mean its symmetric square root.

Symmetric Square Root of $\tilde{\mathbf{P}}^a$

- $\tilde{\mathbf{P}}^a = (k-1)^{-1}\mathbf{W}^a(\mathbf{W}^a)^T$, and $\mathbf{P}^a = (k-1)^{-1}\mathbf{X}^a(\mathbf{X}^a)^T$ follows from

$$\mathbf{P}^a = \mathbf{X}^b \tilde{\mathbf{P}}^a (\mathbf{X}^b)^T.$$

- The use of the symmetric square root to determine \mathbf{W}^a from $\tilde{\mathbf{P}}^a$ is important for two main reasons:
 - it ensures that the sum of the columns of \mathbf{X}^a is zero, so that the analysis ensemble has the correct sample mean
 - it ensures that \mathbf{W}^a depends continuously on $\tilde{\mathbf{P}}^a$; which is crucial in a local analysis scheme, so that neighboring grid points with slightly different matrices $\tilde{\mathbf{P}}^a$ do not yield very different analysis ensembles.
- Also, the symmetric square root minimizes the (mean-square) distance between \mathbf{W}^a and the identity matrix (Ott et al. 2004), though because of the different choice of basis, it does not minimize the same quantity, and thus does not yield the same analysis ensemble

Proof that the sum of the columns of \mathbf{X}^a is zero

- We express this condition as $\mathbf{X}^a \mathbf{v} = \mathbf{0}$, where \mathbf{v} is a column vector of k ones: $\mathbf{v} = (1, 1, \dots, 1)^T$.
- Notice that by (15), \mathbf{v} is an eigenvector of $\tilde{\mathbf{P}}^a$ with eigenvalue $(k - 1)^{-1}$:

$$(\tilde{\mathbf{P}}^a)^{-1} \mathbf{v} = [(k - 1)\mathbf{I} + (\mathbf{Y}^b)^T \mathbf{R}^{-1} \mathbf{Y}^b] \mathbf{v} = (k - 1) \mathbf{v},$$

because the sum of the columns of \mathbf{Y}^b is zero.

- Then by (18), \mathbf{v} is also an eigenvector of \mathbf{W}^a with eigenvalue 1. Since the sum of the columns of \mathbf{X}^b is zero, $\mathbf{X}^a \mathbf{v} = \mathbf{X}^b \mathbf{W}^a \mathbf{v} = \mathbf{X}^b \mathbf{v} = \mathbf{0}$ as desired.

“Weight Vectors”

- Notice that we can form the analysis ensemble first in $\tilde{\mathcal{S}}$ by adding $\bar{\mathbf{w}}^a$ to each of the columns of \mathbf{W}^a ; let $\{\mathbf{w}^{a(i)}\}$ be the columns of the resulting matrix.
- These “weight” vectors specify what linear combinations of the background ensemble perturbations to add to the background mean in order to get the analysis ensemble in model space:

$$\mathbf{x}^{a(i)} = \bar{\mathbf{x}}^b + \mathbf{X}^b \mathbf{w}^{a(i)}, \quad (19)$$

Local Implementation I

- Notice that once the background ensemble has been used to form $\bar{\mathbf{y}}^b$ and \mathbf{Y}^b , it is no longer needed in the analysis, except in (19) to translate the results from \tilde{S} to model space. This point is useful to keep in mind when implementing a local filter that computes a separate analysis for each model grid point.
- In principle, one should form a global background observation ensemble $\mathbf{y}_{[g]}^{b(i)}$ from the global background vectors, though in practice this can be done locally when the global observation operator $H_{[g]}$ uses local interpolation.
- After the background observation ensemble is formed, the analyses at different grid points are completely independent of each other and can be computed in parallel.

Local Implementation II

- The observations chosen for a given grid point will dictate which coordinates of $\mathbf{y}_{[g]}^{b(i)}$ are used to form the local background observation ensemble $\mathbf{y}^{b(i)}$ for that analysis, and the analysis in \tilde{S} will produce the weight vectors $\{\mathbf{w}^{a(i)}\}$ for that grid point
- Thus computing the analysis ensemble $\{\mathbf{x}^{a(i)}\}$ for that grid point using (19) requires only using the background model states at that grid point.
- As long as the sets of observations used at a pair of neighboring grid points overlap heavily, the linear combinations used at the two grid points will be similar, and thus the global analysis ensemble members formed by these spatially varying linear combinations will change slowly from one grid point to the next.

- In other localization approaches (Houtekamer and Mitchell 2001; Hamill et al. 2001, Whitaker and Hamill 2002), the influence of an observation at a particular point on the analysis at a particular model grid point decays smoothly to zero as the distance between the two points increases.
- A similar effect can be achieved here by multiplying the entries in the inverse observation error covariance matrix \mathbf{R}^{-1} by a factor that decays from one to zero as the distance of the observations from the analysis grid point increases.
- This “smoothed localization” corresponds to gradually increasing the uncertainty assigned to the observations until beyond a certain distance they have infinite uncertainty and therefore no influence on the analysis.

Multiplicative Covariance Inflation in the LETKF

- Multiplicative inflation can be performed most easily on the analysis ensemble by multiplying \mathbf{W}^a by an appropriate factor (namely $\sqrt{\rho}$ in order to multiply the analysis covariance by ρ).
- To perform multiplicative inflation on the background covariance instead, one should theoretically multiply \mathbf{X}^b by such a factor, and adjust the background ensemble $\{\mathbf{x}^{b(i)}\}$ accordingly before applying the observation operator H to form the background observation ensemble $\{\mathbf{y}^{b(i)}\}$.
- A more efficient approach, which is equivalent if H is linear, and is a close approximation even for nonlinear H if the inflation factor ρ is close to one, is simply to replace $(k - 1)\mathbf{I}$ by $(k - 1)\mathbf{I}/\rho$ in (15), since $(k - 1)\mathbf{I}$ is the inverse of the background covariance matrix $\tilde{\mathbf{P}}^b$ in the k -dimensional space $\tilde{\mathcal{S}}$.
- One can check that this has the same effect on the analysis mean $\bar{\mathbf{x}}^a$ and covariance \mathbf{P}^a as multiplying \mathbf{X}^b and \mathbf{Y}^b by $\sqrt{\rho}$.

Cookbook I

- The inputs to the analysis are a background ensemble of $m_{[g]}$ -dimensional model state vectors $\{\mathbf{x}_{[g]}^{b(i)} : i = 1, 2, \dots, k\}$, a function $H_{[g]}$ from the $m_{[g]}$ -dimensional model space to the $\ell_{[g]}$ -dimensional observation space, an $\ell_{[g]}$ -dimensional vector $\mathbf{y}_{[g]}^o$ of observations, and an $\ell_{[g]} \times \ell_{[g]}$ observation error covariance matrix $\mathbf{R}_{[g]}$. The subscript g here signifies that these inputs reflect the global model state and all available observations.
1. *Apply $H_{[g]}$ to each $\mathbf{x}_{[g]}^{b(i)}$ to form the global background observation ensemble $\{\mathbf{y}_{[g]}^{b(i)}\}$, and average the latter vectors to get the $\ell_{[g]}$ -dimensional column vector $\bar{\mathbf{y}}_{[g]}^b$. Subtract this vector from each $\{\mathbf{y}_{[g]}^{b(i)}\}$ to form the columns of the $\ell_{[g]} \times k$ matrix $\mathbf{Y}_{[g]}^b$.*

Cookbook II

2. *Average the vectors $\{\mathbf{x}_{[g]}^{b(i)}\}$ to get the $m_{[g]}$ -dimensional vector $\bar{\mathbf{x}}_{[g]}^b$, and subtract this vector from each $\mathbf{x}_{[g]}^{b(i)}$ to form the columns of the $m_{[g]} \times k$ matrix $\mathbf{X}_{[g]}^b$.*

3. *This step selects the necessary data for a given grid point (whether it is better to form the local arrays described below explicitly or select them later as needed from the global arrays depends on one's implementation). Select the rows of $\bar{\mathbf{x}}_{[g]}^b$ and $\mathbf{X}_{[g]}^b$ corresponding to the given grid point, forming their local counterparts: the m -dimensional vector $\bar{\mathbf{x}}^b$ and the $m \times k$ matrix \mathbf{X}^b , which will be used in Step 7. Likewise, select the rows of $\bar{\mathbf{y}}_{[g]}^b$ and $\mathbf{Y}_{[g]}^b$ corresponding to the observations chosen for the analysis at the given grid point, forming the ℓ -dimensional vector $\bar{\mathbf{y}}^b$ and the $\ell \times k$ matrix \mathbf{Y}^b . Select the corresponding rows of $\mathbf{y}_{[g]}^o$ and rows and columns of $\mathbf{R}_{[g]}$ to form the ℓ -dimensional vector \mathbf{y}^o and the $\ell \times \ell$ matrix \mathbf{R} .*

Cookbook III

4. *Compute the $k \times \ell$ matrix $\mathbf{C} = (\mathbf{Y}^b)^T \mathbf{R}^{-1}$.* Since this is the only step in which \mathbf{R} is used, it may be most efficient to compute \mathbf{C} by solving the linear system $\mathbf{R}\mathbf{C}^T = \mathbf{Y}^b$ rather than inverting \mathbf{R} . In some applications, \mathbf{R} may be diagonal, but in others \mathbf{R} will be block diagonal with each block representing a group of correlated observations. As long as the size of each block is relatively small, inverting \mathbf{R} or solving the linear system above will not be computationally expensive.

5. *Compute the $k \times k$ matrix $\tilde{\mathbf{P}}^a = [(k-1)\mathbf{I}/\rho + \mathbf{C}\mathbf{Y}^b]^{-1}$, as in (15).* Here $\rho > 1$ is a multiplicative covariance inflation factor, as described at the end of the previous section. Though trying some of the other approaches described there may be fruitful, a reasonable general approach is to start with $\rho > 1$ and increase it gradually until one finds a value that is optimal according to some measure of analysis quality. Multiplying \mathbf{C} and \mathbf{Y}^b requires less than $2k^2\ell$ operations, while the number of operations needed to invert the $k \times k$ matrix is proportional to k^3 .

Cookbook IV

6. Compute the $k \times k$ matrix $\mathbf{W}^a = [(k-1)\tilde{\mathbf{P}}^a]^{1/2}$, as in (18). Again the number of operations required is proportional to k^3 ; it may be most efficient to compute the eigenvalues and eigenvectors of $[(k-1)\mathbf{I}/\rho + \mathbf{C}\mathbf{Y}^b]$ in the previous step and then use them to compute both $\tilde{\mathbf{P}}^a$ and \mathbf{W}^a .
7. Compute the k -dimensional vector $\mathbf{w}^a = \tilde{\mathbf{P}}^a \mathbf{C}(\mathbf{y}^o - \bar{\mathbf{y}}^b)$, as in (14), and add it to each column of \mathbf{W}^a , forming a $k \times k$ matrix whose columns are the analysis vectors $\{\mathbf{w}^{a(i)}\}$.
8. Multiply \mathbf{X}^b by each $\mathbf{w}^{a(i)}$ and add $\bar{\mathbf{x}}^b$ to get the analysis ensemble members $\{\mathbf{x}^{a(i)}\}$ at the analysis grid point, as in (19).
9. After performing Steps 2–7 for each grid point, the outputs of Step 7 form the global analysis ensemble $\{\mathbf{x}_{[g]}^{a(i)}\}$.

Concluding Remarks About the LETKF

- There are two important unique aspects of the LETKF
 - The state estimation is carried out independently for each grid point. (This aspect was introduced in Ott et al. 2004). More precisely, the state estimation can be carried out independently for each state vector component. (Steps 1-3 and 9).
 - It provides an efficient algorithm for the linear algebra. (Steps 4-8).
- In principle, steps 4-8 can be replaced by any form of the update equations. For instance, Whitaker and Hamill (2007) used the LETKF localization strategy, but with the serial update equations from their 2002 paper. This led to a more efficient implementation of the Whitaker and Hamill (2002) square-root filter scheme. We note that for a large number of observations, this strategy is still less efficient than the LETKF.