6.6 A LOCAL LEAST SQUARES FRAMEWORK FOR ENSEMBLE FILTERING

Jeffrey L. Anderson
NOAA / GFDL and NCAR

1. INTRODUCTION

A framework is developed in which most ensemble Kalman filter methodologies documented to date can be described while still supporting a more general class of ensemble filters. The derivation begins with the nonlinear filtering problem and applies a sequence of simplifying assumptions. The introduction of a joint state / observation space leads to an ability to deal with observations related to the model state variables by non-linear functions. A least squares assumption (equivalent to assuming a local Gaussian relation amongst the prior joint state variables) has been made, sometimes indirectly, in many descriptions of ensemble Kalman filters. Here, that assumption is made explicitly and a significant simplification in the description of the algorithms results. Under the assumptions made here, the ensemble filter problem simplifies to an application of an ensemble filter to a scalar, followed by a sequence of linear regressions.

2. ENSEMBLE FILTERING

This section only discusses what happens at one time at which observations become available. Each ensemble member is integrated forward in time independently using a forecast model between times at which observations are available.

A joint state/observation space is defined by the joint space state vector

$$\mathbf{z} = [\mathbf{x}, \mathbf{h}(\mathbf{x})] = [\mathbf{x}, \mathbf{y}] \qquad (1)$$

where $\mathbf{x}$ is the model state vector, $\mathbf{y} = \mathbf{h}(\mathbf{x})$, referred to as observation variables, is the expected value of the observations available at this time, and $\mathbf{z}$ is a vector of length n + m where n is the number of state variables and m is the number of observations available at this time. Observations are assumed to be selected from a distribution with expected value given by $\mathbf{h}(\mathbf{x})$ and an associated observational error distribution.

Corresponding author address: Jeffrey Anderson, UCAR/MMM, P.O. Box 3000, Boulder, CO 80307-3000; e-mail:jla@ucar.edu

The distribution of the posterior (updated) distribution $\mathbf{z}^u = [\mathbf{x}^u, \mathbf{y}^u]$ can be computed from the prior distribution, $\mathbf{z}^p = [\mathbf{x}^p, \mathbf{y}^p]$, as

$$\mathbf{p}(\mathbf{z}^u) = \mathbf{p}(\mathbf{y}^o \mid \mathbf{z}^p)\mathbf{p}(\mathbf{z}^p) \,/\, \textit{normalization} \qquad (2)$$

where $\mathbf{y}^o$ is an m-vector of the observed values available at this time. (2) implies that subsets of observations with independent (observational error) distributions can be assimilated sequentially. Let $\mathbf{y}^o$ be composed of s subsets of observations, $\boldsymbol{y}^o = \{\boldsymbol{y}^o_1, \boldsymbol{y}^o_2, \ldots, \boldsymbol{y}^o_s\}$, where the distribution of the observational errors for observations in subset i is independent of the distribution for the observations in subset j, for $i \neq j$. Then

$$\boldsymbol{p}(\boldsymbol{y}^o \,|\, \boldsymbol{z}^p) = \prod_{i=1}^{s} \boldsymbol{p}(\boldsymbol{y}^o_i \,|\, \boldsymbol{z}^p). \qquad (3)$$

If the observational error covariance matrix, **R**, is not diagonal, a singular value decomposition can be performed on **R**. The prior joint state ensembles can be projected onto the singular vectors and the assimilation can proceed. The updated state vectors can be projected back to the original state space. Given the application of this SVD, a mechanism for sequential assimilation of scalar observations implies no loss of generality for observations with arbitrary Gaussian error distributions.

In ensemble methods for solving (2), information about the prior distribution of the state variables, $\mathbf{x}^p$, is available as a sample of size N produced by N applications of a prediction model. A sample of the prior observation vector, $\mathbf{y}^p$, can be created by applying the forward observation operator, $\mathbf{h}$, to each ensemble sample of $\mathbf{x}^p$.

Define the joint state space forward operations operator for a single observation as the order 1 x n+1 linear operator $\mathbf{H} = [0, 0, ..., 0, 1]$. The expected value of the observation can be calculated by applying $\mathbf{H}$ to the joint state vector, $\mathbf{z}$. The conversion of the possibly non-linear $\mathbf{h}$ to the linear

H is a primary motivation for applying ensemble filters in the joint state space.

The updated probability for the *marginal* distribution of the observation joint state variable, y, can be formed from eq. (2)

$$p_y(y^u) = \boldsymbol{p}(y^o \mid y^p)\,\boldsymbol{p}_y(y^p)/(norm) \qquad (4)$$

where the subscript on the probability densities indicates a marginal probability on the observation variable, y. The 1-D problem for this marginal distribution can be solved by many methods.

This suggests a partitioning of the assimilation of an observation into two parts. The first determines updated ensemble members for the observation variable y given the observation, y^o. An increment, Δy_i, is computed for each ensemble member, $y_i^u = y_i^p + \Delta y_i$, i = 1, ..., N where N is the ensemble size.

Given increments for the observation variable, the second step computes corresponding increments for each ensemble sample of each state variable, $\Delta x_{i,j}$. This requires assumptions about the prior relationship between the joint state variables. The assumption used here is that the prior distribution is Gaussian (or a sum of Gaussians for complete generality). This is equivalent to assuming a least squares fit (local least squares fit) to the prior ensemble members summarizes the relationship between the joint state variables.

Figure 1 depicts an example in which there is a single state variable, x. The observation variable, y, is related to x by the operator h which is nonlinear in the figure. Increments for each ensemble sample of y have been computed. The corresponding increments for x are then computed by a global least squares fit (linear regression) so that

$$\Delta x_i = \frac{\sigma_{x,y}}{\sigma_{y,y}} \Delta y_i \ . \qquad (5)$$

This is just a statistical linearization and inversion of the operator h.

The linearization can also be done locally (Fig. 2) by computing local estimates of covariance for each ensemble member, for instance, by only using a set of nearest neighbors to compute sample covariance. Figure 2 shows an idealized form of nearest neighbor linearization in which only a single closest ensemble member is used to compute the statistical linearization. When x is functionally related to y as in Fig. 2, local linearization methods can give significantly enhanced performance when h is strongly non-linear over the prior ensemble range of y.

This two step method can be extended trivially to problems with arbitrary numbers of state variables. Least squares fits can be made to compute the increments for each state variable independently by regression:

$$\Delta x_{i,j} = \frac{\sigma_{xj,y}}{\sigma_{y,y}} \Delta y_i . \qquad (6)$$

All relevant information about the prior covariance of the model state variables, **x,** needed to compute increments is contained in the correlation of the individual scalar state variables with the observation variable y.

When the state variable being updated and the observation variable are not functionally related as is generally the case in multivariate models, the use of local linearizations can be more problematic. Figure 3 shows an example where state variable x_1 is being updated by an observation, y^o. The expected value of the observation is $y = h(x_2)$, where x_2 is a second state variable, here moderately correlated with x_1. In this case, the linear regression for x_1 performs a statistical linearization in the presence of noise. Using large (global) regressions is useful to filter out this noise. On the other hand, using local linearizations can help to resolve more of the structure of h. Applying local regressions that are based on too few ensemble members can lead to disastrous over-fitting behavior as demonstrated by the application of an idealized single nearest neighbor linearization in Figure 3. Trade-offs in choosing local versus global linearizations are an important part of tuning ensemble filters for improved performance.

3. RELATION TO ENSEMBLE KALMAN FILTERS

A variety of ensemble Kalman filters have been described in the literature. The two most common of these, the perturbed observations ensemble Kalman filter (EnKF) and the Ensemble Adjustment Kalman Filter (or square root Kalman filter) can be recast in the two step framework outlined in the previous section preserving answers to machine precision. First, update increments are computed for each ensemble sample of the observation variable using scalar versions of the traditional algorithms. Eq. (6) is then used to solve for

the increments, $\Delta x_{j,i}$, for each state variable in turn in terms of Δy_i by linear regression.

Some implications about the computational complexity of ensemble filtering can be drawn from (6). First, there is no need to compute the prior covariance among the model state variables or the complete updated covariance, Σ^u. Second, once the observational variables are updated, the increments for the state variables depend only on ratios of prior (co)variances. Multiplication of the prior covariance matrix by a covariance inflation factor as is done in many existing ensemble Kalman filter implementations does not impact the solution to (6) although it does impact the observation variable increments.

4. ADDITIONAL METHODS FOR UPDATING OBSERVATIONAL VARIABLE ENSEMBLES

Once update increments for the observation variable are computed by one of the following methods, the rest of the joint state variables can be updated by linear regression using (6).

4.1 *A Kernel filter*

If the prior distribution of the observation variable may have significant non-Gaussian structure, a kernel method may be useful for computing the update increments. One simple example of kernel methods is the Fukunaga method applied in one dimension. In this algorithm, the prior distribution is represented as a sum of Gaussians with identical variance but different means. The means are the individual prior ensemble samples and the variance is the prior sample variance multiplied by a scaling factor. The product of a prior expressed as a sum of Gaussians and a Gaussian observational distribution is equal to the sum of the products of the individual prior Gaussians and the observational Gaussian. In the most naive application of this method, an updated ensemble can be generated from this continuous representation by randomly sampling the sum of Gaussians.

This kernel method can be extended by allowing more general kernels. For instance, kernels with different means and different variances can be used following a variety of techniques like the class of nearest neighbor methods. In addition, kernels from the class of 'generalized' Gaussians can lead to a variety of related kernel algorithms.

4.2 *Quadrature Product Methods*

Update methods based directly on 'quadrature' solutions to (4) can be used to find increments for observation variables. One implementation of such a method begins by computing a continuous approximation to the prior distribution from the ensemble sample; again, kernel methods are an example. Quadrature methods can then be used to divide the real line into intervals over which the product in the numerator of (4) is computed to approximate the updated distribution. An appropriate method can be used to sample this updated distribution to generate new ensemble methods.

5. CONCLUSIONS

A local least squares framework for ensemble filtering has been derived leading to a two-step ensemble filtering update procedure when a new observation becomes available. Step one is to compute update increments for each ensemble member of a prior estimate of the value of the observation in a joint state / observation space. This can be done using various algorithms including the perturbed observation ensemble Kalman filter and the ensemble adjustment Kalman filter. Other update methods, for instance a kernel filter, extend beyond the Kalman filter context and can be referred to more generally as ensemble filters.

Step two computes increments for prior ensemble members for each state variable in turn using the prior ensemble sample to do a linear regression of each state variable on the observation variable. Increments for a state variable are computed by multiplying the corresponding observation variable increment by the prior covariance of the state and observation variable and dividing by the prior variance of the observation variable.

Deriving ensemble filters in this two step context has many advantages. First, it is computationally more efficient than previous descriptions of ensemble Kalman filter algorithms. The cost is dominated by computation of the prior sample cross covariance of the observation and state variables and the variance of the observation variables. A second advantage is that more elaborate and expensive ensemble update methods can be applied because they need only be applied in a scalar fashion to the observation variables. A final advantage is that it is easier to understand differences between various filtering algorithms. Differences need only be explored in a scalar context making relative features of, for instance, the EnKF and EAKF easier to understand.

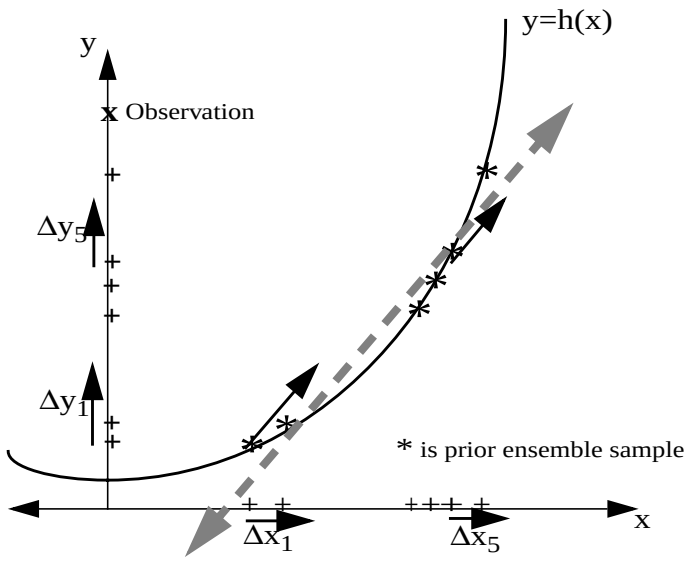

Figure 1: An idealized representation showing the relation between update increments for a state variable, x, and an observation variable, y, for a five member ensemble represented by asterisks. The projection of the ensemble on the x and y axes is represented by '+' and the observation, y^o is represented by 'x'. In this case, y is functionally related to x by h. The dashed grey curve shows a global least squares fit to the ensemble members. Update increments for ensemble members 1 and 5 for y are shown along with corresponding increments for the ensemble as a whole (vectors parallel to least squares fit) and for the x ensemble.

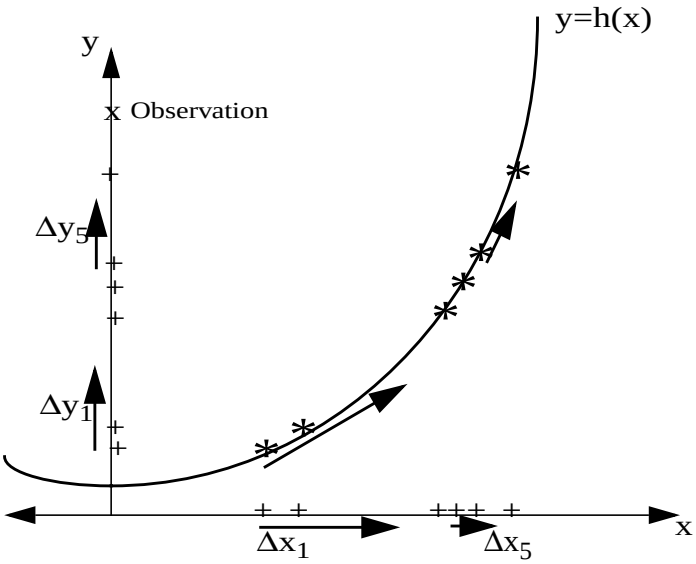

Figure 2: As in Figure 1 but showing the application of local least squares fits, in this case using only the nearest neighbor in y, to compute the updates for x given the updates for y.

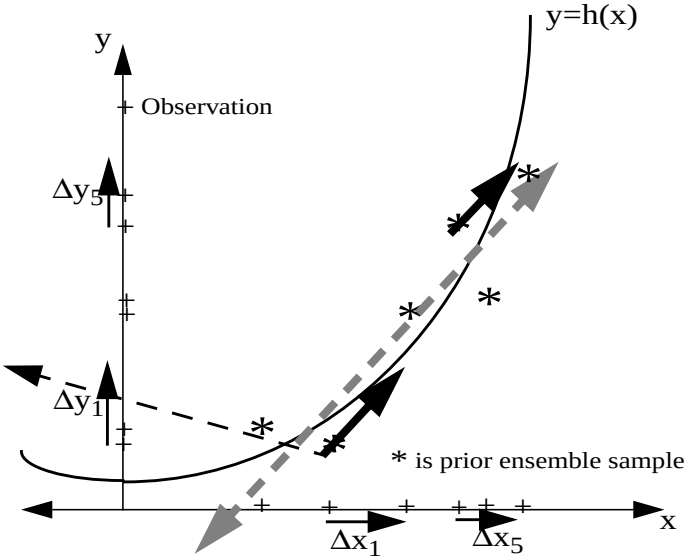

Figure 3: As in Figure 1, but now y = h(x2), where x2 is a second state variable that is moderately correlated with x. The thin dashed vector demonstrates the hazard of using local least squares fits when the observation variable y and the state variable x are not functionally related.