Assimilation Techniques (5): Approximate Kalman Filters and Singular Vectors April 2001

By **Mike Fisher**

European Centre for Medium-Range Weather Forecasts.

Table of contents

1 . Introduction

2 . Why is the Kalman filter impractical for very large systems?

3 . The ensemble Kalman filter

4 . Subspaces, projections and Hessian singular vectors

5 . The ECMWF reduced-rank Kalman filter

5.1 The subspace

5.2 Theinner product

5.3 The background cost function

6 . Examples

REFERENCES

1. Introduction

For a perfect, linear model and linear observation operators, both 4dVar and the Kalman filter give the same values for the model variables at the end of the 4dVar assimilation window, provided that both systems start with the same covariance matrices at the beginning of the window. The fundamental difference between the Kalman filter and 4dVar is that the former explicitly evolves the covariance matrix, whereas the covariance evolution in 4dVar is implicit. This means that when we come to perform another cycle of analysis, the Kalman filter provides us with both a model state (background) and its covariance matrix. 4dVar does not provide the covariance matrix.

The aim of this lecture is to describe some approaches to Kalman filterering for very large systems. Particular emphasis is placed on describing the ECMWF "reduced rank" Kalman filter. This is under development as a possible replacement for the current 4dVar system.

2. Why is the Kalman filter impractical for very large systems?

The Kalman filter augments the usual equations for the analysis and propagation of the model state:

$$\begin{aligned} \mathrm{x}_a &= \mathrm{x}_b + \mathrm{P}^b\mathrm{H}^{\mathrm{T}}(\mathrm{R} + \mathrm{H}\mathrm{P}^b\mathrm{H}^{\mathrm{T}})^{-1}(\mathrm{y} - \mathrm{H}\mathrm{x}_b) \\ \mathrm{x}_f &= M(\mathrm{x}_a) \end{aligned} \tag{1}$$

with equations which analyse and propagate the covariance matrix:

$$\begin{aligned} \mathrm{P}^a &= \mathrm{P}^b - \mathrm{P}^b\mathrm{H}^{\mathrm{T}}(\mathrm{R} + \mathrm{H}\mathrm{P}^b\mathrm{H}^{\mathrm{T}})^{-1}\mathrm{H}\mathrm{P}^b \\ \mathrm{P}^f &= \mathrm{M}\mathrm{P}^a\mathrm{M}^{\mathrm{T}} + \mathrm{Q} \end{aligned} \tag{2}$$

(Note: To avoid confusing integer subscripts denoting analysis cycle, the background and forecast states and covariance matrices have been distinguished in Eqs. (1) and (2) by subscripts and superscripts, b and f. Of course, the forecast quantities x_f and P^f form the background state x_b and covariance matrix P^b for the next analysis cycle. The matrix Q represents the effect of model error.)

For large systems, the computational expense of the Kalman filter is in the analysis and the propagation of the covariance matrix. There are two expensive operations. First, to calculate the analysis error covariance matrix, we must invert the matrix $\mathrm{R} + \mathrm{H}\mathrm{P}^b\mathrm{H}^{\mathrm{T}}$. Second, the covariance matrix has to be propagated in time according to the dynamics of the tangent linear model. This requires the application of the tangent linear model to each column and the adjoint model to each row of the (typically $10^6 \times 10^6$) covariance matrix. In additional to the prohibitive computational expense of these operations, the covariance matrices are too large to store in current computer memories.

A full Kalman filter has been run at ECMWF for a three-level T21 quasi-geostrophic model. This is more or less the largest system for which a full Kalman filter is computationally feasible on current computers. The computational cost rises rapidly with the dimension of the problem since higher resolution increases both the computational cost of the model, and the number of model integrations which are required.

For the forseeable future, therefore, the full Kalman filter will remain too expensive for use as a meteorological analysis sytem for numerical weather prediction. We are forced to approximate. In particular, we must find ways to reduce the number of integrations of the tangent linear model by several orders of magnitude.

3. THE ENSEMBLE KALMAN FILTER

The ensemble Kalman filter (Evensen 1994, Houtekamer and Mitchell 1998) takes a statistical approach to the solution of the Kalman filter equations for the covariance matrices of analysis and background error. The idea of the method is to generate a statistical sample of analyses. This is done by running the analysis system several times for a given date, each time using backgrounds which differ by an amount characteristic of background error, and observations which have been perturbed by adding random noise drawn from the distribution of observation error. (The backgrounds are produced by running short forecasts from each member of the preceding ensemble of analyses.)

The differences between the analyses form a statistical sample of analysis error, which may be used to estimate the covariance matrix of analysis error, without the need for expensive matrix inversions:

$$\mathrm{P}^a \approx \langle \mathrm{x}_a(\mathrm{x}_a)^{\mathrm{T}} \rangle \tag{3}$$

where $\langle . \rangle$ represents an average over the ensemble.

By running a short forecast from each ensemble analysis, we get an ensemble of backgrounds which give an estimate of the covariance matrix of background error:

$$\mathrm{P}^b \approx \langle \mathrm{x}_b(\mathrm{x}_b)^{\mathrm{T}} \rangle \tag{4}$$

The estimated covariance matrices have rank equal to the size of the ensemble (i.e. several orders of magnitude smaller than the dimension of the matrix). This rank deficiency means that if we try to use the estimated background covariance matrix directly in the analysis, we will generate analysis increments which lie strictly in the space spanned by the ensemble members. One solution to this problem is to split the analysis increment into a part which projects onto the subspace spanned by the ensemble members, and a part which is orthogonal. Then, we may use the statistically estimated covariance matrix for the projected part of the analysis increment, and a static covariance matrix for the orthogonal part.

A second way to avoid the problem of rank-deficiency of the covariance matrix is to modify the covariance matrix in a way which increases its rank. Houtekamer and Mitchell 2000 suggest modifying the estimated covariances by removing spurious covariances at large distance. This has the effect of greatly increasing the rank of the estimated covariance matrix.

The ensemble Kalman filter has several attractive features. It is very well suited to modern parallel computers, since the members of the analysis ensemble are essentially independent and may therefore be run simultaneously. The method works by trying to diagnose the actual covariance matrix of background error rather than by explicitly propagating an approximate analysis error covariance matrix. This may be an advantage. On the other hand, the random errors in the statistically-estimated covariances decrease only as the square-root of the ensemble size. Also, random vectors do not provide an optimal subspace for explaining forecast errors.

4. SUBSPACES, PROJECTIONS AND HESSIAN SINGULAR VECTORS

The most general way to reduce the number of tangent linear model integrations required to propagate the covariance matrix, is to choose some low-dimensional subspace, and to propagate the projection of the covariance matrix onto the subspace. (In the case of the Ensemble Kalman Filter, the subspace is defined by the ensemble of analyses.) For the rest of the phase space, we should do something sensible but cheap. For example, we could decide to use a static (flow-independent) covariance matrix.

The most general way to choose a subspace is to choose a set of K vectors s_k which span it. However, this is not enough. We must also decide what we mean by projection onto the subspace. That is, we must choose an inner product $\langle .,. \rangle$.

Given any vector x, we can define a part which projects onto the subspace:

$$\mathrm{x}_\mathrm{S} = \sum_{k=1}^{K} \alpha_k \mathrm{s}_k \tag{5}$$

and a part which is orthogonal:

$$\mathrm{x}_{\bar{\mathrm{S}}} = \mathrm{x} - \mathrm{x}_\mathrm{S} \tag{6}$$

The coefficients of the projection α_k are completely determined by the requirement that $\mathrm{x}_{\bar{\mathrm{S}}}$ is orthogonal to the subspace according to our chosen inner product (i.e. that $\langle \mathrm{x}_{\bar{\mathrm{S}}}, \mathrm{s}_k \rangle = 0$ for $k = 1 \ldots K$).

Particularly interesting for the analysis is the set of vectors $\mathrm{s}_k(t_0)$ at the analysis time t_0, which evolve into the leading eigenvectors $\mathrm{s}_k(T)$ of the forecast error covariance matrix P^f at some future time T. (Typically, we might take T to be 48 hours after the analysis time.) These vectors are interesting, because for a fixed number of vectors, the leading eigenvectors of P^f are the vectors which account for a maximum variance of forecast error. The vectors $\mathrm{s}_k(t_0)$ define precisely the directions in which we want to do a good job of analysis if we are to minimize the fore-

cast error two days later.

Now, we want the vectors $s_k(T)$ to be eigenvectors of the forecast error covariance matrix. But, it is meaningless to perform an eigendecomposition of a matrix whose elements are not all of the same dimension. We therefore need to non-dimensionalize things. Let us define non-dimensional vectors:

$$\hat{s}_k(T) = W^{1/2} s_k(T) \tag{7}$$

where $W^{1/2}$ is a matrix which non-dimensionalizes the variables.

The corresponding non-dimensional forecast error covariance matrix is:

$$W^{1/2} P^f (W^{1/2})^{\mathrm{T}} \tag{8}$$

We want the non-dimensional vectors $\hat{s}_k(T)$ to be the leading eigenvectors of the non-dimensional forecast error covariance matrix. That is:

$$W^{1/2} P^f (W^{1/2})^{\mathrm{T}} \hat{s}_k(T) = \lambda_k \hat{s}_k(T) \tag{9}$$

In terms of the dimensional vectors, we have:

$$P^f W s_k(T) = \lambda_k s_k(T) \tag{10}$$

where $W = (W^{1/2})^{\mathrm{T}} W^{1/2}$.

Note that the eigenvector problem depends on our choice of non-dimensionalization. The choice amounts to deciding a way to compare the relative magnitudes of forecast errors. Different choices give different eigenvectors.

Having chosen a set of vectors at time T, we can turn our attention to the analysis time t_0. We want small perturbations in the directions of the vectors at analysis time $s_k(t_0)$, to evolve into perturbations in the directions of the vectors $s_k(T)$. That is, we want:

$$s_k(T) = M_{t_0 \to T} s_k(t_0) \tag{11}$$

where $M_{t_0 \to T}$ represents the tangent linear model. Substituting this into the eigenvector equation gives:

$$P^f W M_{t_0 \to T} s_k(t_0) = \lambda_k M_{t_0 \to T} s_k(t_0) \tag{12}$$

Now suppose that model error is negligible over the period t_0 to T. In this case, the forecast error covariance matrix is related to the analysis error covariance matrix P^a through:

$$P^f = M_{t_0 \to T} P^a M^{\mathrm{T}}{}_{t_0 \to T} \tag{13}$$

Substituting this into Eq. (12) gives:

$$M_{t_0 \to T} P^a M^{\mathrm{T}}{}_{t_0 \to T} W M_{t_0 \to T} s_k(t_0) = \lambda_k M_{t_0 \to T} s_k(t_0) \tag{14}$$

If we now cancel the leading $M_{t_0 \to T}$ from both sides of the equation, and then multiply by $(P^a)^{-1}$, we get the

following generalized eigenvector equation:

$$\mathrm{M}^{\mathrm{T}}{}_{t_0 \to T}\mathrm{W}\,\mathrm{M}_{t_0 \to T}\;\mathrm{s}_k(t_0) \;=\; \lambda_k(\mathrm{P}^a)^{-1}\mathrm{s}_k(t_0) \tag{15}$$

This is an equation we can solve to determine a few of the leading eigenvectors using a variant of the Lanczos algorithm (Barkmeijer *et al.* 1998). The algorithm requires the ability to calculate the products of the matrices on both sides of the equation with arbitrary vectors. This is clearly possible for the left hand side of the equation. It requires one integration of the tangent linear and adjoint models. On the right hand side, we use the fact that in a variational analysis system, $(\mathrm{P}^a)^{-1}$ is equal to the Hessian matrix of the analysis cost function, J''. So, for any vector x, we can calculate the product $(\mathrm{P}^a)^{-1}\mathrm{x}$ as

$$(\mathrm{P}^a)^{-1}\mathrm{x} \;=\; J''\mathrm{x} = \frac{1}{\varepsilon}(\nabla J(\mathrm{x}_0 + \varepsilon\mathrm{x}) - \nabla J(\mathrm{x}_0)) \tag{16}$$

The vectors $\mathrm{s}_k(t_0)$ are known as Hessian singular vectors (Barkmeijer *et al.* 1999).

Another interpretation of Hessian singular vectors is that they are the vectors which maximize the ratio:

$$\lambda_k \;=\; \frac{\mathrm{s}_k^{\mathrm{T}}(T)\mathrm{W}\mathrm{s}_k(T)}{\mathrm{s}_k^{\mathrm{T}}(t_0)(\mathrm{P}^a)^{-1}\mathrm{s}_k(t_0)} \tag{17}$$

(Another way to derive the generalized eigenvector equation, 15, is to take the gradient of λ_k with respect to $\mathrm{s}_k(t_0)$, and to note that if λ_k is at a maximum, then its gradient is zero.)

The numerator, $\mathrm{s}_k^{\mathrm{T}}(T)\mathrm{W}\mathrm{s}_k(T)$ is a measure of the "size" of the vector $\mathrm{s}_k(T)$. The denominator, is a measure of the likelihood of an analysis error $\mathrm{s}_k(t_0)$, since for Gaussian errors:

$$\log(\mathrm{prob}(\mathrm{s}_k(t_0))) \;=\; \mathrm{const} - \frac{1}{2}\mathrm{s}_k^{\mathrm{T}}(t_0)(\mathrm{P}^a)^{-1}\mathrm{s}_k(t_0) \tag{18}$$

In other words, the Hessian singular vectors are the vectors which for a given initial likelihood grow to maximum "size" at time T (where "size" is measured by W).

5. THE ECMWF REDUCED-RANK KALMAN FILTER

5.1 The subspace

The ECMWF reduced-rank Kalman filter is an approximate Kalman filter which uses Hessian singular vectors to define a subspace. The filter is based on the observation that if we define

$$\mathrm{z}_k(t_0) \;=\; \frac{1}{\lambda_k}\mathrm{M}^{\mathrm{T}}{}_{t_0 \to T}\mathrm{W}\,\mathrm{M}_{t_0 \to T}\;\mathrm{s}_k(t_0) \tag{19}$$

then, by equation 15, we have $\mathrm{z}_k(t_0) \;=\; (\mathrm{P}^a)^{-1}\mathrm{s}_k(t_0)$. That is, the vector $\mathrm{z}_k(t_0)$ gives the action of the inverse analysis error covariance matrix on the Hessian singular vector at initial time.

More generally, for any time $t > t_0$, we may define vectors at time t: $\mathrm{s}_k(t) \;=\; \mathrm{M}_{t_0 \to t}\;\mathrm{s}_k(t_0)$ and $\mathrm{z}_k(t) \;=\; (1/\lambda_k)\mathrm{M}^{\mathrm{T}}{}_{t \to T}\mathrm{W}\,\mathrm{M}_{t \to T}\;\mathrm{s}_k(t)$. It is easy to show that these vectors satisfy

$$z_k(t) = (P^f(t))^{-1} s_k(t) \tag{20}$$

where $P^f(t)$ is the forecast error covariance matrix at time t.

In particular, by chosing t to be the time at which the background for the next analysis cycle is valid, we get a set of vectors $s_k(t)$ which we may use to define a subspace, and a set of vectors $z_k(t)$ which define the action of the inverse of the next cycle's background error covariance matrix on the subspace.

Once the Hessian singular vectors are known, the vectors $z_k(t)$ and $s_k(t)$ are easily calculated.

5.2 Theinner product

Having chosen a set of vectors with which to define a subspace, we must now consider which inner product to use to define projection onto the subspace. This will allow us to partition any vector $x(t)$ into a part $x_S(t)$ which lies in the subspace spanned by the Hessian singular vectors, and a part $x_{\bar{S}}(t)$ which is orthogonal to the subspace.

Consider how the the initial orthogonal component evolves in time. At time T, we have:

$$x_{\bar{S}}(T) = M_{t \to T} x_{\bar{S}}(t) \tag{21}$$

We would like $x_{\bar{S}}(T)$ to be orthogonal to the vectors $s_k(T)$ with respect to some suitable inner product. The obvious inner product is defined by the matrix W. The vectors $s_k(T)$ are orthogonal in the sense that for $j \neq k$ we have:

$$(s_j(T))^{\mathrm{T}} W (s_k(T)) = 0 \tag{22}$$

Unfortunately, this does not lead to a convenient inner product for use at the analysis time.

However, the Hessian singular vectors satisfy a second orthogonality condition. For any time $t_0 \leq t \leq T$ and $j \neq k$, it can be shown that:

$$(s_j(t))^{\mathrm{T}} (P^f(t))^{-1} (s_k(t)) = 0 \tag{23}$$

This allows us to define an inner product:

$$\langle x_{\bar{S}}(t), s_k(t) \rangle \equiv (x_{\bar{S}}(t))^{\mathrm{T}} (P^f(t))^{-1} (s_k(t)) \tag{24}$$

Suppose we require $\langle x_{\bar{S}}(t), s_k(t) \rangle = 0$ for $k = 1 \ldots K$. Now, $x_{\bar{S}}(T) = M_{t \to T} x_{\bar{S}}(t)$ and $P^f(T) = M_{t \to T} P^f(t) M_{t \to T}$. So, at time T, we will have:

$$(x_{\bar{S}}(T))^{\mathrm{T}} (P^f(T))^{-1} (s_k(T)) = 0 \tag{25}$$

Thus, the evolved vector $x_{\bar{S}}(T)$ is orthogonal to the evolved Hessian singular vectors with respect to the inner product defined by the inverse of the evolved covariance matrix.

In fact, we do not know the covariance matrix $P^f(t)$, so we cannot use it to define an inner product. However, we do have an approximation to it, in the form of the the static background error covariance matrix B. The ECMWF reduced rank Kalman filter therefore uses the inner product:

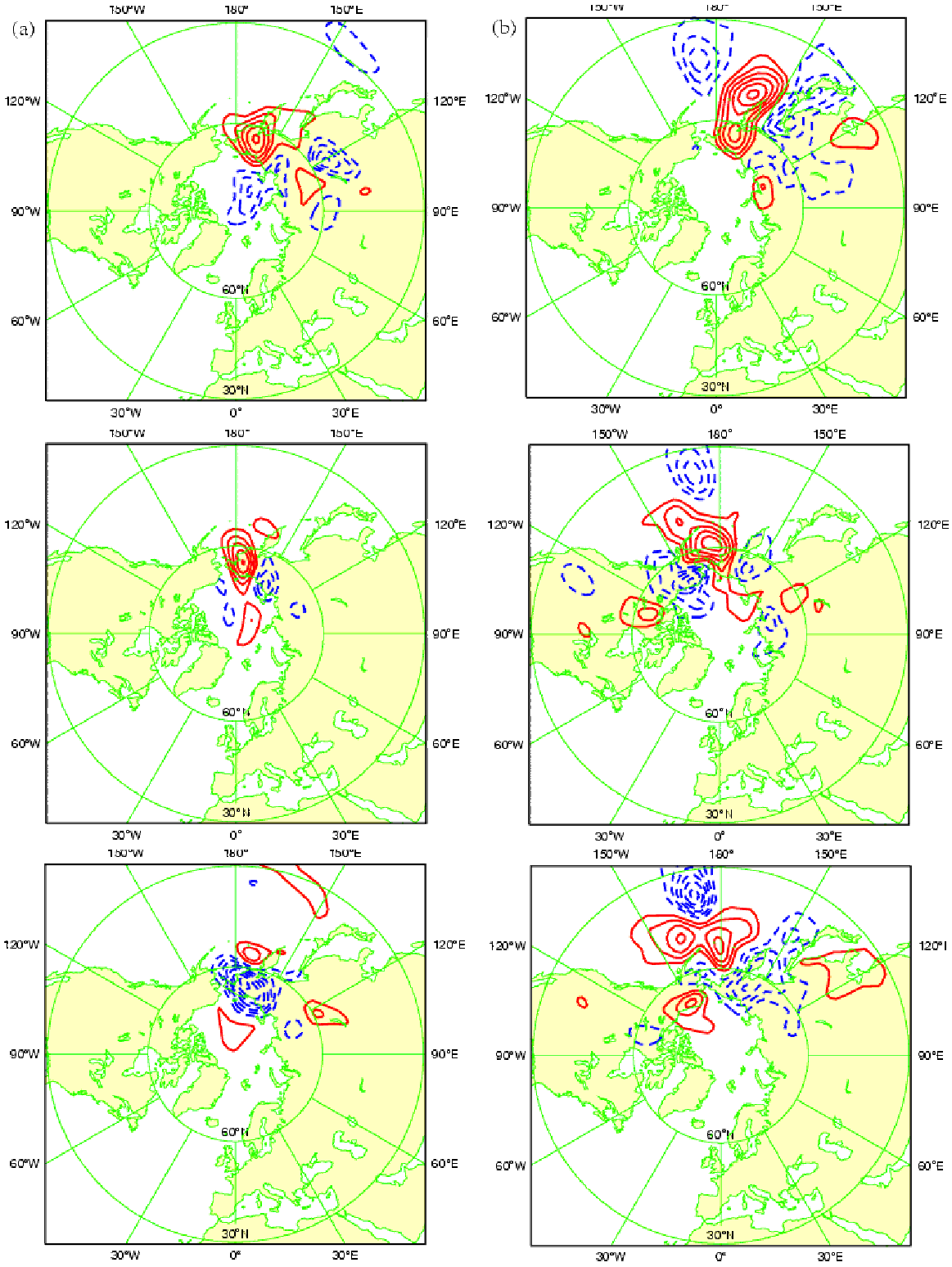

Figure 1. Streamfunction on model level 39 (approx. 500 hPa) for the 3 leading Hessian singular vectors for 03z 15 October 1999. (a) Vectors at initial time. (b) Vectors at final time, 48 hours later.

$$\langle x_{\bar{S}}(t), s_k(t) \rangle \equiv (x_{\bar{S}}(t))^{\mathrm{T}} \mathrm{B}^{-1} (s_k(t)) \tag{26}$$

5.3 The background cost function

The ECMWF reduced rank Kalman filter uses 4dVar to perform the analysis, but modifies the background cost function to use a flow dependent covariance matrix for the subspace defined by the Hessian singular vectors. If we partition the background departure δx into its projection onto the supspace and its orthogonal component, the modified background cost function may be written as:

$$J_b = \frac{1}{2}(\delta x_S)^{\mathrm{T}} (P^f)^{-1} \delta x_S + (\delta x_S)^{\mathrm{T}} \mathrm{B}^{-1} \delta x_{\bar{S}} + \frac{1}{2}(\delta x_{\bar{S}})^{\mathrm{T}} \mathrm{B}^{-1} \delta x_{\bar{S}} \tag{27}$$

Now, we have chosen to use the inner product defined by Eq. (26). Hence, the second term on the right hand side vanishes. The background cost function is therefore positive definite. To evaluate the first term, note that:

$$(P^f)^{-1} \delta x_S = \sum_{k=1}^{K} \alpha_k (P^f)^{-1} s_k(t) = \sum_{k=1}^{K} \alpha_k z_k(t) \tag{28}$$

where α_k are the projection coefficients. (In fact, the ECMWF reduced rank Kalman filter implements the modified background cost function in a somewhat different, but algebraically equivalent way. The details of the implementation will not be described here. They are described by Fisher 1998.)

6. EXAMPLES

Fig. 1 shows the streamfunction at model level 39 (approximately 500 hPa) for the leading three Hessian singular vectors for initial time 03z 15 October 1999. A 4dVar Hessian was used, and the optimization time $T - t_0$ was 48 hours. The matrix W was equivalent to a dry energy inner product.

In the lecture on 4dVar, it was shown that the increment generated by an observation at the start of the assimilation window is completely determined by the static background error covariance matrix. For the reduced-rank Kalman filter, the increment for such an observation is determined by the modified background cost function, and therefore includes a flow-dependent component. Fig. 2 illustrates that this is indeed the case. Fig. 2 (a) shows a cross-section of the temperature increment generated in 4dVar by a pair of geopotential height observations at 500 hPa at the beginning of the assimilation window. The slight asymmetry of the increment on the left of the figure is due to the effect of orography (vertical correlations are defined with respect to model levels, whereas the vertical coordinate in Fig. 2 (a) is pressure). The increments generated by the same observations in the reduced-rank Kalman filter are shown in Fig. 2 (b). The increment to the left of the figure is clearly modified. The observation in this case was close to a front, as is shown by a cross-section of potential temperature (Fig. 3). The negative contours extend along the frontal zone, and the positive contours are tilted westward with height.

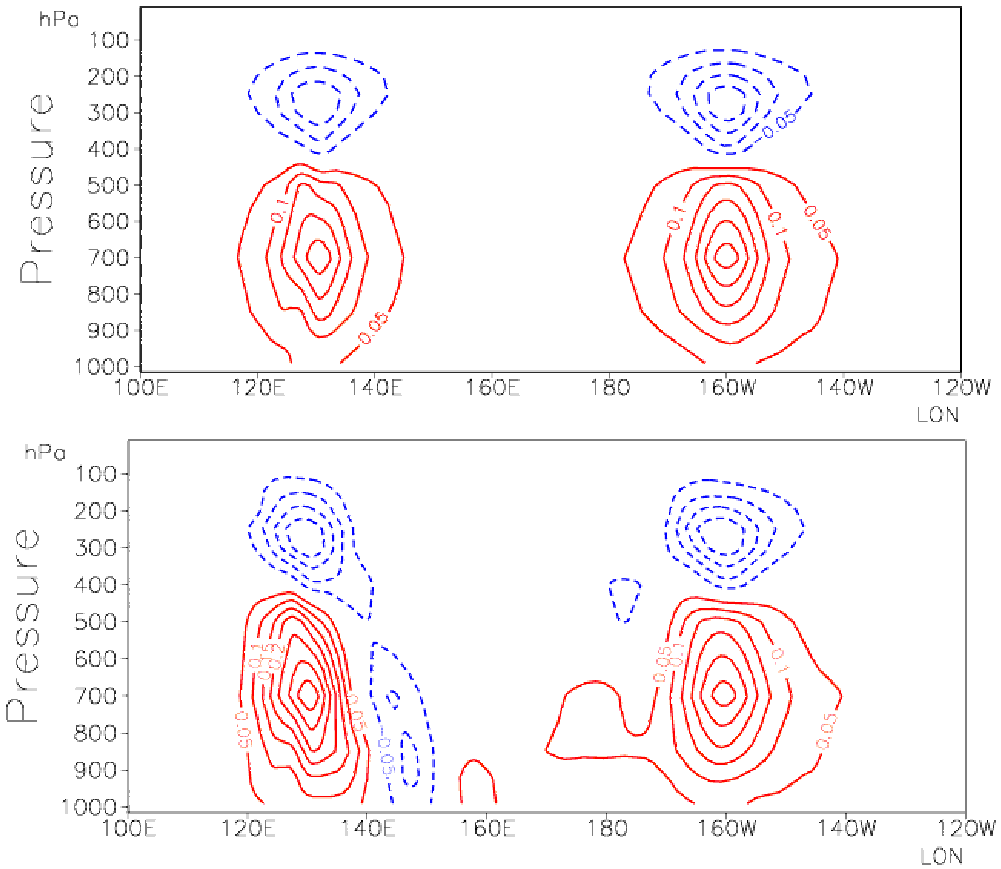

Figure 2. Temperature increments along latitude 40 N generated by two 500 hPa height observations at the beginning of the assimilation window (a) in 4dvar, (b) in the reduced-rank Kalman filter.

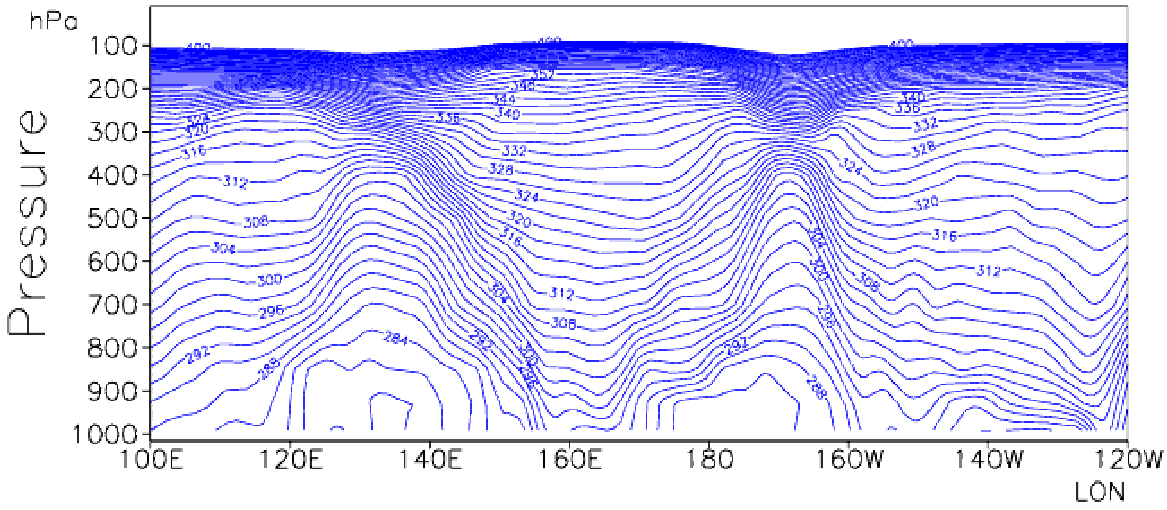

Figure 3. Cross-section of potential temperature along latitude 40 N corresponding to Fig. 2 . Values larger than 400 K have not been contoured.

REFERENCES

Evensen G., 1994, Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. J. Geophys. Res., 99(C5), 10143-10162.

Houtekamer P.L. and H.L. Mitchell, 1998, Data assimilation using an ensemble Kalman filter technique. Mon. Wea. Rev. 126, 796-811.

Houtekamer P.L. and H.L. Mitchell, 2000, A sequential Ensemble Kalman Filter for atmospheric data assimilation, submitted to Mon. Wea. Rev.

Barkmeijer J., M. Van Gijzen and F. Bouttier, 1998, Singular vectors and estimates of the analysis error covariance metric. Q. J. Roy. Meteor. Soc. 124, 1695-1713.

Barkmeijer J., R. Buizza and T.N. Palmer, 1999, 3D-Var Hessian singular vectors and their potential use in the ECMWF Ensemble Prediction System, Q. J. Roy. Meteor. Soc. 125, 2333-2351.

Fisher M., 1998, Development of a simplified Kalman filter, ECMWF Technical memorandum 260.