TECHNICAL MEMORANDUM

220

Estimating the covariance matrices of analysis and forecast error in variational data assimilation

M. Fisher and P. Courtier

Research Department

August 1995

This paper has not been published and should be regarded as an Internal Report from ECMWF.
Permission to quote from it should be obtained from the ECMWF.

European Centre for Medium-Range Weather Forecasts
Europäisches Zentrum für mittelfristige Wettervorhersage
Centre européen pour les prévisions météorologiques à moyen

Estimating the covariance matrices of analysis and forecast error in variational data assimilation
Document presented to the 24th Session of the SAC, Sept 1995

---

1. INTRODUCTION

Data assimilation attempts to combine observed and predicted values of the state variables of a system in an optimal way to produce the best possible estimate of the true state. The optimal combination of observed and predicted quantities requires knowledge of the corresponding covariance matrices of observational and short-term forecast errors. The former matrix is relatively easy to specify, since it depends primarily on instruments whose characteristics are well known and whose errors are typically spatially uncorrelated. Estimation of the covariance matrix of short-term prediction error is more difficult.

For a linear system, the Kalman filter (*Kalman*, 1960) produces the Best Linear Unbiased Estimate (BLUE) of the state of the system at the final time of an observing period. The extended Kalman filter (see *Ghil and Malanotte-Rizzoli*, 1991) is the natural generalization of the Kalman filter to weakly nonlinear systems. It may be divided into a forecast step and an analysis step. In the forecast step,

$$x^f(t_i) = M(t_i,t_{i-1})x^a(t_{i-1}) \tag{1a}$$

$$\mathbf{P}^f(t_i) = \mathbf{M}(t_i,t_{i-1})\mathbf{P}^a(t_{i-1})\mathbf{M}(t_i,t_{i-1})^T + \mathbf{Q}(t_{i-1}) \tag{1b}$$

the analyzed state, $\mathbf{x}^a(t_{i-1})$, at time t_{i-1}, and the covariance matrix of analysis error, $\mathbf{P}^a(t_{i-1})$, are propagated in time. Here, M represents a discrete model for the evolution of the system; $\mathbf{M}$ represents a linearization of M; $\mathbf{Q}$ represents the covariance matrix of model error; and $\mathbf{T}$ denotes matrix transposition. (The notation adopted in this paper is that proposed by *Ide et al.*, 1995.) In the analysis step,

$$x^a(t_i) = x^f(t_i) + K_i(y_i - H_i x^f(t_i)) \tag{2a}$$

$$P^a(t_i) = (I-K_iH_i)P^f(t_i) \tag{2b}$$

the state of the system is estimated as a combination of observed and predicted values, and the forecast error covariance matrix is modified to give the covariance matrix of analysis error, $\mathbf{P}^a(t_i)$. Here, H_i is an observation operator which estimates the observations, $\mathbf{y}_i$, from the predicted state; $\mathbf{H}_i$ is a linearization of H_i; and $\mathbf{K}_i$ is the Kalman gain matrix.

In current numerical weather prediction systems, the dimensions of the covariance matrices of analysis and prediction error are of the order of $10^6 \times 10^6$. This is far too large for the matrices to be represented explicitly, or for the equations 1b and 2b to be solved exactly. A number of simplifying approximations must be made. Typically, correlations are approximated by fixed functions which are isotropic and largely homogeneous.

Equation 1b is replaced by a simple inflation of the variances of analysis error, which in turn are estimated using a simplified version of equation 2b. Recently, several authors (for example, *Cohn and Todling*, 1995 and *Ehrendorfer and Tribbia*, 1995) have discussed approximations for equation 1b which incorporate the tangent linear dynamics.

It is well known (e.g. *Rabier et al.*, 1992) that, for the linear case and a perfect model, the Kalman filter and four dimensional variational data assimilation (4d-Var) are equivalent in that they produce the same estimate of the state of a system at the final time of an assimilation period. However, in 4d-Var the covariance matrices of analysis and forecast error are not explicitly known. In a cycling analysis system, it is necessary to determine an approximation for the covariance matrix of forecast error, since this provides the covariance matrix of background error for the next analysis cycle. It is also useful to be able to estimate the covariance matrix of analysis error.

The mathematical tools provided by the introduction of adjoint techniques to data assimilation, together with the mathematical clarity of the variational formulation of the data assimilation problem, provide the opportunity for significantly improving on currently used methods for approximating the covariance matrices of analysis and short-term forecast errors. This paper discusses a few methods for determining approximate solutions of equations 1b and 2b in the context of variational data assimilation, with an emphasis on methods which are sufficiently computationally efficient to be of practical use in an operational system.

2. OVERVIEW

Subject to the approximations that error growth is linear and that model error is small, the short-term forecast error covariance matrix is given approximately by

$$\mathbf{P}=\mathbf{M}\mathbf{A}\mathbf{M}^{T} \tag{3}$$

where $\mathbf{A}$ is the covariance matrix for errors in the analysis which provided initial conditions for the forecast and $\mathbf{M}$ represents the tangent linear model integrated for the period of the forecast.

Ehrendorfer and Tribbia (1995) point out that the eigenvectors of a covariance matrix provide the most efficient approximation to the matrix in the sense that, for an approximation based on a given number of vectors, they account for a maximum fraction of the variance. Ehrendorfer and Tribbia demonstrated, in a simple model, how an approximation to $\mathbf{P}$ may be generated from its leading eigenvectors. However, their algorithm requires explicit knowledge of the symmetric square root of $\mathbf{A}$, and this is unlikely to be available in an operational data assimilation system. Direct determination of the eigenpairs of $\mathbf{P}$, for example by applying a Lanczos algorithm to $\mathbf{M}\mathbf{A}\mathbf{M}^{\mathrm{T}}$, is also not possible in the variational context, since $\mathbf{A}$ is not known.

An algorithm to determine eigenpairs of $\mathbf{P}$ for a variational data assimilation system may be achieved by considering the generalized eigenvector equation

$$\mathbf{M}^{T}\mathbf{M}x = \lambda\mathbf{A}^{-1}x. \tag{4}$$

Multiplying through by $\mathbf{M}\mathbf{A}$ gives $\mathbf{M}\mathbf{A}\mathbf{M}^{\mathrm{T}}(\mathbf{M}x)=\lambda(\mathbf{M}x)$. Thus, the vectors $(\mathbf{M}x)$ are eigenvectors of $\mathbf{P}$.

In section 2, it is demonstrated that in incremental variational data assimilation, $\mathbf{A}^{-1}$ is equal to the Hessian matrix of the cost function. If the cost function is quadratic (as it is in the incremental formulation), then its gradient is a linear function of $\mathbf{x}$ and satisfies $\mathbf{A}^{-1}\mathbf{x}=\nabla J(\mathbf{x})-\nabla J(\mathbf{0})$. Multiplication of a vector by $\mathbf{A}^{-1}$ may therefore be achieved without explicit knowledge of $\mathbf{A}^{-1}$ by means of a single gradient calculation. Multiplication of a vector by $\mathbf{M}^T\mathbf{M}$ requires an integration of the tangent linear model followed by an integration of its adjoint. Thus, the operators required to evaluate both sides of equation 3 are available in incremental variational assimilation. Methods exist (e.g. Scott, 1981) to find eigenvectors of a generalized eigenvector problem given the ability to multiply arbitrary vectors by the matrices on either side of the problem (in this case $\mathbf{M}^T\mathbf{M}$ and $\mathbf{A}^{-1}$). Thus, solution of equation 3 represents a practical algorithm for determining eigenpairs of $\mathbf{P}$.

A significant factor for an operational assimilation system is the computational cost of the algorithms used. The method outlined above is at least as expensive as a 4d-Var analysis. Any algorithm to determine the eigenvectors of the generalized problem must require at least one gradient calculation and one integration of the tangent linear model and its adjoint for each eigenvector it determines. Each generalized eigenvector must also be propagated by an integration of the tangent linear model to give the vectors ($\mathbf{Mx}$), which are the eigenvectors of $\mathbf{P}$. Even if the cost is reduced by decreasing the spatial resolution of the model, the algorithm is still likely to require a larger than acceptable proportion of the computer time available.

In practice, algorithms which directly solve the generalized eigenvector problem are computationally expensive, requiring many more matrix multiplications for each eigenpair than are required to determine eigenpairs of an ordinary eigenvalue problem. A more efficient approach is to consider the ordinary eigenvalue problem, $\mathbf{MAM}^T\mathbf{x}=\lambda\mathbf{x}$. Since the analysis error covariance matrix is not available as an operator in variational assimilation, it must be approximated. Three computationally efficient algorithms for approximating $\mathbf{A}$ in variational assimilation are presented in section 3. The algorithms are compared in the context of a simplified analysis system in section 4. In section 5, the most promising of the three algorithms is evaluated in the ECMWF 3d-Var analysis system.

All the approximations of the analysis error covariance matrix presented in this paper take the form of low-rank corrections to the background error covariance matrix, $\mathbf{B}$. ($\mathbf{B}$ is an approximation of the covariance matrix of forecast error for the preceding analysis cycle.) As an alternative to calculating eigenvectors of $\mathbf{P}$, approximations of the forecast error covariance matrix may be achieved by applying a modified low-rank correction to an approximation of $\mathbf{MBM}^T$. For two of the approximate analysis error covariance matrices presented in this paper, the modification of the correction matrix may be achieved with a few integrations of the tangent linear model. For the third method, under restricted circumstances, it may be achieved without integrations of the tangent linear model or its adjoint using information generated during the minimization. Approximations to $\mathbf{MBM}^T$ may be generated by replacing $\mathbf{B}$ with a low-rank matrix of the form $\mathbf{WW}^T$, where $\mathbf{W}$ is a rectangular matrix comprising a few vectors. Two such approximations are discussed in section 6.

3. THREE METHODS FOR APPROXIMATING THE COVARIANCE MATRIX OF ANALYSIS ERROR

Variational data assimilation minimises a cost function, J, which measures the closeness of fit of the analysis to a set of observations, and to an a priori (background) estimate $\mathbf{x}^b$ of the field being analyzed. Specifically, the analysis is defined as the vector $\mathbf{x}^a$ which minimises

$$J(x) = \frac{1}{2}(x-x^b)^T\mathbf{B}^{-1}(x-x^b) + \frac{1}{2}(H(x)-y^o)^T\mathbf{R}^{-1}(H(x)-y^o) \quad 5$$

Here, H is an operator which, when applied to $\mathbf{x}$, produces an estimate of the observation vector, $\mathbf{y}^o$. $\mathbf{B}$ is the covariance matrix of $(\mathbf{x}^t - \mathbf{x}^b)$, where $\mathbf{x}^t$ is the true value of $\mathbf{x}$, and $\mathbf{R}$ is the covariance matrix of $(H(\mathbf{x}^t)-\mathbf{y}^o)$.

In incremental variational data assimilation, the nonlinear operator $H(\mathbf{x})$ is approximated by the linear operator $H(\mathbf{x}^b) + \mathbf{H}(\mathbf{x}-\mathbf{x}^b)$, where the matrix $\mathbf{H}$ represents the tangent linear operator of H. A consequence of this approximation is that the Hessian matrix of the cost function is independent of $\mathbf{x}$ and is given by

$$J'' = \mathbf{B}^{-1} + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}. \quad (6)$$

The gradient of the cost function at $\mathbf{x}^a$ is zero. In the incremental formulation, this gives

$$\mathbf{B}^{-1}(x^a-x^t-(x^b-x^t)) + \mathbf{H}^T\mathbf{R}^{-1}(\mathbf{H}(x^a-x^t) + \mathbf{H}x^t-y^o)=0, \quad (7)$$

which may be rearranged to give

$$(\mathbf{B}^{-1}+\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H})\ (x^a-x^t) = \mathbf{B}^{-1}(x^b-x^t)+\mathbf{H}^T\mathbf{R}^{-1}(\mathbf{H}x^t-y^o). \quad (8)$$

Multiplying each side of this equation to the right by its transpose, taking the ensemble average, and noting that the covariance matrices of $(\mathbf{x}^b-\mathbf{x}^t)$ and $(\mathbf{H}\mathbf{x}^t-\mathbf{y}^o)$ are $\mathbf{B}$ and $\mathbf{R}$ respectively, gives the following expression for the covariance matrix $\mathbf{A}$ of errors in the analysis. (It is assumed that background and observation errors are uncorrelated.)

$$(\mathbf{B}^{-1}+\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H})\ \mathbf{A}(\mathbf{B}^{-1}+\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H})^T = (\mathbf{B}^{-1}+\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H})^T. \quad (9)$$

Hence

$$\mathbf{A} = (\mathbf{B}^{-1}+\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H})^{-1}. \quad (10)$$

Comparing equations 10 and 6 shows that in incremental variational data assimilation, the analysis error covariance matrix is equal to the inverse of the Hessian matrix of the cost function. This classical relationship is exploited in all of the methods for estimating $\mathbf{A}$ presented in this paper.

3.1. The Randomization method

Rabier and Courtier (1992) used the randomization method to estimate the diagonal elements of the Hessian matrix for a 4d-Var analysis. The method depends on the equivalence between the Hessian matrix and the covariance matrix of the gradient, which may be derived as follows.

Consider the function

$$\hat{J}^o(x) = \frac{1}{2}(H(x)-\hat{y})\mathbf{R}^{-1}(H(x)-\hat{y})$$

where $\hat{\mathbf{y}}=\boldsymbol{H}(\mathbf{x}^b)-\xi$, and where ξ is a random variable with zero mean and covariance matrix $\mathbf{R}$.

The gradient of $\hat{J}^o$ at $\mathbf{x}=\mathbf{x}^b$ is $\mathbf{H}^T\mathbf{R}^{-1}\xi$. This is a random variable with covariance matrix

$$\mathbf{H}^T\mathbf{R}^{-1}\overline{\xi\xi^T}\mathbf{R}^{-1}\mathbf{H}. \tag{12}$$

But, the covariance matrix of ξ is $\mathbf{R}$, so the covariance matrix of the gradients of $\hat{J}^o$ is simply $\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}$. Substituting for $\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}$ in equation 10 gives

$$\mathbf{A}^{-1} = \mathbf{B}^{-1}+\overline{(\nabla\hat{J}^o)(\nabla\hat{J}^o)^T}. \tag{13}$$

An approximation to the inverse of the analysis error covariance matrix may be calculated by replacing the covariance matrix of the gradients in equation 13 by the covariance matrix of a small sample of gradients. The resulting expression for $\mathbf{A}^{-1}$ is easily inverted using the Shermann-Morrison-Woodbury formula (*Shermann and Morrison*, 1949).

3.2. The Lanczos method

The Lanczos algorithm (*Lanczos*, 1950) provides a method for calculating some of the eigenvectors and eigenvalues of a matrix $\mathbf{G}$, without requiring explicit knowledge of the elements of $\mathbf{G}$. At the n^{th} iteration of the algorithm, a rectangular matrix $\mathbf{Q}_n$ of orthonormal vectors and an n x n tridiagonal matrix $\mathbf{T}_n$ are generated, satisfying

$$\mathbf{G}\mathbf{Q}_n = \mathbf{Q}_n\mathbf{T}_n. \tag{14}$$

The eigenvalues of $\mathbf{T}_n$ are approximations to the eigenvalues of $\mathbf{G}$, and premultiplying the eigenvectors of $\mathbf{T}_n$ by $\mathbf{Q}_n$ gives approximate eigenvectors of $\mathbf{G}$.

It is well known (e.g. *Paige and Saunders*, 1975) that there is a close connection between the Lanczos algorithm and conjugate gradient minimisation. It can be shown (see appendix A) that the vectors which comprise the matrix $\mathbf{Q}_n$ are proportional to the gradient vectors generated during conjugate gradient minimisation with exact line searches of a quadratic form with Hessian matrix $\mathbf{G}$. The elements of $\mathbf{T}_n$ may also be calculated from quantities generated during the minimisation. Applying this result to incremental variational data assimilation allows simultaneous minimisation of the cost function and derivation of the eigenpairs of the analysis error covariance matrix.

If minimisation is carried out in terms of the transformed control vector $\chi=\mathbf{L}^{-1}\mathbf{x}$, where $\mathbf{L}\mathbf{L}^T=\mathbf{B}$, then the Hessian matrix is equal to the sum of the identity matrix and a matrix of rank less than or equal to the dimension of the vector of observations. Thus, most of the eigenvalues of the Hessian are equal to one and the Hessian matrix in terms of χ is well approximated by

$$\mathbf{I}+\sum_{k=1}^{M}(\lambda_k-1)\boldsymbol{v}_k\boldsymbol{v}_k^T \tag{15}$$

where λ_k and $\mathbf{v}_k$ are selected eigenvalues and eigenvectors of the Hessian.

This matrix is easily inverted and transformed to model variable space, to give an approximation to the analysis error covariance matrix in terms of $\mathbf{x}$:

$$\mathbf{A} \sim \mathbf{B} + \sum_{k=1}^{M} (\lambda_k^{-1} - 1)(\mathbf{L}\mathbf{v}_k)(\mathbf{L}\mathbf{v}_k)^T. \tag{16}$$

The combined Lanczos and conjugate gradient algorithm requires that the cost function be strictly quadratic. This restriction may preclude its use for the main analysis in an operational context. In particular, it prohibits the adoption of variational quality control of observations via non-Gaussian statistics, as described by *Dharssi et al.*, 1992. However, one may envisage a system in which a preliminary (possibly reduced resolution) analysis with a quadratic cost function is performed using the combined Lanczos and conjugate gradient algorithm. This analysis would provide a good initial point and a preconditioner for the main analysis, as well as providing an estimate of the analysis error covariance matrix.

3.3. The BFGS method

The Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm (*Broyden*, 1969) was developed in the context of quasi-Newton minimization methods. For unconstrained minimization problems, it is generally regarded as the best method of approximating the inverse of the Hessian (*Dennis and Moré*, 1977). The algorithm generates a sequence of approximations to the inverse of the Hessian matrix of a cost function given a sequence of vectors $\mathbf{x}_k$ and gradients $\nabla J(\mathbf{x}_k)$.

Let $\mathbf{s}_k = \mathbf{x}_k - \mathbf{x}_{k-1}$, and $\mathbf{y}_k = \nabla \mathbf{J}(\mathbf{x}_k) - \nabla \mathbf{J}(\mathbf{x}_{k-1})$. The sequence of approximations to the inverse of the Hessian matrix is generated by the update formula

$$\mathbf{G}^{(k)} = \left(\mathbf{I} - \frac{s_k \times y_k}{\langle y_k, s_k \rangle}\right) \mathbf{G}^{(k-1)} \left(\mathbf{I} - \frac{y_k \times s_k}{\langle y_k, s_k \rangle}\right) + \frac{s_k \times s_k}{\langle y_k, s_k \rangle} \tag{17}$$

Here, $\langle \cdot, \cdot \rangle$ denotes an inner product, and $\times$ denotes the corresponding tensor product (that is $(\mathbf{a} \times \mathbf{b})$ is the rank one matrix such that for any vector $\mathbf{x}$, $(\mathbf{a} \times \mathbf{b})\mathbf{x} = \langle \mathbf{b}, \mathbf{x} \rangle \mathbf{a}$). The starting matrix, $\mathbf{G}^{(0)}$, should be a positive definite approximation to the inverse of the Hessian matrix. In practice, $\mathbf{G}^{(0)}$ is usually chosen to be a diagonal matrix. An efficient algorithm exists to multiply an arbitrary vector by $\mathbf{G}^{(k)}$, without requiring explicit construction of the matrix (*Nocedal*, 1980).

The inverse of $\mathbf{G}^{(k)}$ is also easily expressed as an update formula:

$$(\mathbf{G}^{(k)})^{-1} = (\mathbf{G}^{(k-1)})^{-1} + \frac{y_k \times y_k}{\langle y_k, s_k \rangle} - \frac{(\mathbf{G}^{(k-1)})^{-1}(s_k \times s_k)(\mathbf{G}^{(k-1)})^{-1}}{\langle s_k, (\mathbf{G}^{(k-1)})^{-1} s_k \rangle}. \tag{18}$$

The BFGS approximation to the inverse of the Hessian matrix has a number of desirable properties (*Dennis and Moré*, 1977). If $\mathbf{G}^{(k-1)}$ is a nonsingular symmetric positive definite matrix, and $\langle \mathbf{y}_k, \mathbf{s}_k \rangle > 0$ (a condition

which can be enforced without significant loss of efficiency) then $\mathbf{G}^{(k)}$ is also nonsingular, symmetric and positive definite. $\mathbf{G}^{(k)}$ also satisfies the quasi-Newton equations $\mathbf{G}^{(k)}\mathbf{y}_j = \mathbf{s}_j$ for j=1,...,k. It can be shown that if $\mathbf{y}_k$ and $\mathbf{s}_k$ are generated using the quasi-Newton algorithm with exact line searches, then $\mathbf{G}^{(k)}$ converges to the inverse of the Hessian matrix in a finite number of steps.

Usually, minimization of the cost function in variational assimilation is not performed in terms of physical variables. Rather, a change of variable is introduced to improve the rate of convergence of the minimization problem. If the transformation from the space in which minimization is performed to the space of physical variables is denoted by multiplication by a matrix $\mathbf{U}$, then an approximation to the covariance matrix of analysis error for physical variables is given by $\mathbf{UG}^{(k)}\mathbf{U}^*$. This approximation may be generated using the following transformation property of the BFGS update formula. Let $\mathbf{U}$ be nonsingular, $\mathbf{U}^*$ denote the adjoint of $\mathbf{U}$ with respect to the inner product, $\langle\cdot,\cdot\rangle$, and write equation 17 as $\mathbf{G}^{(k)}=F(\mathbf{G}^{(k-1)},\mathbf{y}_k,\mathbf{s}_k)$, then

$$\mathbf{UG}^{(k)}\mathbf{U}^* = F(\mathbf{UG}^{(k-1)}\mathbf{U}^*, (\mathbf{U}^*)^{-1}\mathbf{y}_k, \mathbf{Us}_k). \tag{19}$$

Thus, by transforming the vectors $\mathbf{y}_k$ and $\mathbf{s}_k$, and providing a suitable starting matrix, the vectors generated during minimization may be used to generate an approximate analysis covariance matrix in terms of physical variables. Another application of the transformation property of the BFGS update formula will be considered in section 8.

The transformation property is easily proved given the following identities (see appendix B):

Identity 1: $\mathbf{(a\times b)A \equiv a\times(A^*b)}$

Identity 2: $\mathbf{A(a\times b) \equiv (Aa)\times b}$

Let $\mathbf{U}$ be a nonsingular matrix. Multiplying equation 17 to the left by $\mathbf{U}$ and to the right by $\mathbf{U}^*$, gives

$$\mathbf{UG}^{(k)}\mathbf{U}^* = \left(\mathbf{I} - \mathbf{U}\frac{\mathbf{s}_k\times\mathbf{y}_k}{\langle y_k,s_k\rangle}\mathbf{U}^{-1}\right)\mathbf{UG}^{(k-1)}\mathbf{U}^*\left(\mathbf{I} - (\mathbf{U}^*)^{-1}\frac{\mathbf{y}_k\times\mathbf{s}_k}{\langle y_k,s_k\rangle}\mathbf{U}^*\right) + \mathbf{U}\frac{\mathbf{s}_k\times\mathbf{s}_k}{\langle y_k,s_k\rangle}\mathbf{U}^*. \tag{20}$$

Define $\hat{\mathbf{s}}_k=\mathbf{Us}_k$ and $\hat{\mathbf{y}}_k=(\mathbf{U}^*)^{-1}\mathbf{y}_k$, then $\langle\hat{\mathbf{s}}_k,\hat{\mathbf{y}}_k\rangle = \langle\mathbf{s}_k,\mathbf{y}_k\rangle$. Applying identities 1 and 2 gives

$$\mathbf{UG}^{(k)}\mathbf{U}^* = \left(\mathbf{I} - \frac{\hat{\mathbf{s}}_k\times\hat{\mathbf{y}}_k}{\langle\hat{y}_k,\hat{s}_k\rangle}\right)\mathbf{UG}^{(k-1)}\mathbf{U}^*\left(\mathbf{I} - \frac{\hat{\mathbf{y}}_k\times\hat{\mathbf{s}}_k}{\langle\hat{y}_k,\hat{s}_k\rangle}\right) + \frac{\hat{\mathbf{s}}_k\times\hat{\mathbf{s}}_k}{\langle\hat{y}_k,\hat{s}_k\rangle}, \tag{21}$$

which is simply the BFGS update formula with $\mathbf{s}_k$, $\mathbf{y}_k$, $\mathbf{G}^{(k-1)}$ and $\mathbf{G}^{(k)}$ replaced respectively by $\hat{\mathbf{s}}_k$, $\hat{\mathbf{y}}_k$, $\mathbf{UG}^{(k-1)}\mathbf{U}^*$ and $\mathbf{UG}^{(k)}\mathbf{U}^*$.

Quasi-Newton minimisation does not require explicit access to the elements of $\mathbf{G}^{(k)}$. In variational data assimilation, however, it is useful to be able to determine some of these elements (in particular the diagonal elements, which are the variances of analysis error). Specific elements of $\mathbf{G}^{(k)}$ may be calculated by rewriting equation 17, using identities 1 and 2, in the form

$$\mathbf{G}^{(k)}=\mathbf{G}^{(k-1)}+(s_k\times s_k)\left(\frac{1}{\langle y_k,s_k\rangle}+\frac{\langle y_k,\mathbf{G}^{(k-1)}y_k\rangle}{\langle y_k,s_k\rangle^2}\right)$$
$$-\frac{1}{\langle y_k,s_k\rangle}\left(s_k\times(\mathbf{G}^{(k-1)*}y_k)+(\mathbf{G}^{(k-1)}y_k)\times s_k\right), \tag{22}$$

and noting that if the inner product is defined as

$$\langle a,b\rangle = a^T\mathbf{Q}b, \tag{23}$$

then the matrix element $(\mathbf{s}\times\mathbf{y})_{i,j}$ is simply $(\mathbf{s})_i(\mathbf{Q}\mathbf{y})_j$.

In an operational data assimilation system, the vectors $\mathbf{s}_k$ and $\mathbf{y}_k$ are likely to be large enough that it is not practical to store more than a few pairs of vectors. An approximate inverse Hessian may be generated from the most recent pairs of vectors only. However, it is useful to retain some information from earlier pairs. *Gilbert and Lemaréchal* (1989) proposed using a diagonal starting matrix which is updated at each iteration using a diagonalized form of the BFGS update formula. The ability to determine the diagonal elements of $\mathbf{G}^{(k)}$, combined with the transformation property, suggests an alternative method. For each k, the formula may be applied in terms of the transformed variable $\mathbf{z}=\Sigma_{k-1}^{-1}\mathbf{x}$, where Σ_{k-1} is the estimate of the standard deviations of analysis error, in terms of physical quantities, calculated at the previous iteration. The starting matrix, $\mathbf{G}_z^{(0)}$ should approximate the correlation matrix of analysis error. The approximate inverse Hessian in terms of $\mathbf{x}$ is given by $\Sigma_{k-1}\mathbf{G}^{(k)}\Sigma_{k-1}^*$. This approach provides a form of dynamic scaling which is particularly useful if an approximation to the standard deviations of analysis error is sought, since it is precisely this information which is retained from earlier iterations. In the simplified analysis problem, the diagonal scaling of Gilbert and Lemaréchal proved significantly more effective for the purpose of minimizing the cost function, but the scaling described here proved more effective for estimating the standard deviations of analysis error (see later). The two approaches may be combined, with the diagonal scaling of Gilbert and Lemaréchal used for the minimization, and the vectors generated during the minimization used together with the scaling described here to estimate the standard deviation of analysis error.

4. RESULTS FROM A SIMPLIFIED 3D-VAR ANALYSIS SYSTEM

The three methods described above have been evaluated for a simple, univariate implementation of incremental 3d-Var on a cyclic one-dimensional domain consisting of 256 equally-spaced grid points. The background error covariance matrix was chosen to have homogeneous correlations but spatially varying variances. Observations were defined at 120 locations with a spatial density similar to the longitudinal distribution of the mid-latitude radiosonde network. For each location, three observations were specified. All observation errors were uncorrelated and all had standard deviation equal to the mean standard deviation of background error.

The correlation structure function and the locations of the observations used are shown in figure 1. Figures 2a and 2b show the standard deviations (in arbitrary units) of background and analysis error for the system, together with approximations generated using the techniques described above. For figure 2a, 35 random gradients were used for the randomization method, and 35 iterations were performed for the BFGS and Lanczos methods. For figure 2b, 10 random gradients were calculated and 10 iterations were performed. Within each figure, the computational cost of the three approximations was similar.

The Lanczos algorithm generates an upper bound on $\|(J'')\mathbf{v}-\theta\mathbf{v}\|/\|\mathbf{v}\|$ for each approximate eigenpair, $(\mathbf{v},\theta)$. The Lanczos approximation to the covariance matrix of analysis error was generated using eigenpairs for which this bound was smaller than 10% of the approximate eigenvalue. The number of eigenpairs satisfying the criterion was 5 after 10 iterations and 31 after 35 iterations.

For the BFGS method, minimization was carried out in terms of the variable χ using the diagonal scaling of Gilbert and Lemaréchal. $\mathbf{G}^{(0)}$ was set to the identity matrix in χ-space (i.e. to the covariance matrix of background error).The approximate standard deviations marked "inner scaling" in figure 2 were also generated using the diagonal scaling of Gilbert and Lemaréchal, whereas those marked "outer scaling" were generated from the same vectors, but used the further transformation to the variable $\mathbf{z}$ described in section 3.3.

Figure 2a demonstrates that even though the number of vectors used is much smaller than the dimension of the problem, all three methods succeed in approximating the main features of the analysis error variances. After 35 iterations (fig. 2a), both versions of the BFGS method tend to underestimate the variances of analysis error. The outer scaling method successfully represents the small scale fluctuations, whereas inner scaling produces an approximation which is too smooth and which significantly underestimates the peaks of analysis error standard deviation. After 10 iterations (fig.2b), the inner scaling method produces a good approximation in regions where the standard deviation of analysis error is small, but a poor approximation elsewhere. The outer scaling method gives a more accurate approximation of peak standard deviations, but overestimates smaller values.

The randomization and Lanczos methods overestimate the standard deviations of analysis error. After 35 iterations, the Lanczos approximation is clearly better than either the randomization or BFGS approximations. After 10 iterations, however, the BFGS method is the most successful. The relatively poor performance of the Lanczos algorithm in this case may reflect the fact that only five vectors are used to generate the approximation, whereas the BFGS and randomization methods extract useful information from ten gradients.

The approximate analysis error variances produced by the Lanczos algorithm are equal to the background variances minus the sum of the squares of elements of a set of vectors. As the number of vectors is increased, each additional vector can only decrease the estimated variance. Since the approximation converges to the true variances of background error, it follows that the approximate analysis error variances lie strictly between the true values and the corresponding variances of background error. For exact line searches, the BFGS method also converges to the true analysis error covariance matrix after a finite number of steps (*Dennis and Moré*, 1977). The randomization method produces unbiased estimates of the elements of the Hessian, but the convergence of these estimates is slow, with the expected error proportional to the square root of the number of vectors. The elements of the analysis error covariance matrix also converge slowly. Furthermore, they are biased.

Figure 3 shows contours of the differences between the approximate and true correlation matrices of analysis error for the various methods for 35 iterations of minimization and 35 random vectors. For the BFGS method, differences are shown for both of the scaling techniques described in section 3.3. The contour interval is 0.1 and solid contours indicate positive differences. For clarity, the zero contour is not shown.

For the BFGS method, outer scaling gives a better estimate of the long range correlations than the diagonal (inner) scaling of Gilbert and Lemaréchal, whereas inner scaling is better at estimating the correlation structure near the diagonal. However, the bands of positive and negative differences near the diagonal for both implementations of the BFGS method indicate that the correlation structure is too broad. This reflects the use of the background error correlation matrix as the starting matrix $\mathbf{G}^{(0)}$. Randomization is better than the BFGS method at capturing the structure near the diagonal, but produces spurious correlations for large separations. The Lanczos method successfully estimates the correlation structure at all ranges.

5. RESULTS FROM THE FULL ANALYSIS SYSTEM

The combined Lanczos and conjugate gradient algorithm has been implemented in the ECMWF 3d-Var assimilation system (*Courtier, et al.*, 1993). As described in section 3.2, it is desirable to employ a change of variable which transforms the background error covariance matrix to the identity matrix. This is not possible for the current formulation of the background error term in the ECMWF system. The results presented here have therefore been performed with a modified system for which the background error covariance matrix is defined as $\mathbf{B}=\mathbf{L}\mathbf{L}^{\mathrm{T}}$ and where the change of variable implied by $\mathbf{L}$ is

$$\chi=\begin{pmatrix} \mathbf{C}^{-\frac{1}{2}}\Sigma_{\zeta}^{-1}\mathbf{W}_{R\zeta}^{\frac{1}{2}} & 0 & 0 \\ 0 & \mathbf{C}^{-\frac{1}{2}}\Sigma_{D}^{-1}\mathbf{W}_{GD}^{\frac{1}{2}} & 0 \\ \mathbf{C}^{-\frac{1}{2}}\Sigma_{P}^{-1}(\mathbf{W}_{RP}^{\frac{1}{2}}-\mathbf{W}_{GP}^{\frac{1}{2}})\wp & 0 & \mathbf{C}^{-\frac{1}{2}}\Sigma_{P}^{-1}\mathbf{W}_{GP}^{\frac{1}{2}} \end{pmatrix}\begin{pmatrix}\zeta\\ D\\ P\end{pmatrix}. \tag{24}$$

Here, ζ, D and P are respectively the vorticity, divergence and linearized geopotential components of $\mathbf{x}$. The matrices, $\mathbf{C}$, Σ_{ζ}, $\mathbf{W}_{R\zeta}$, etc. are unmodified from the standard formulation. $\mathbf{C}$ is a correlation matrix; Σ_{ζ}, Σ_{D} and Σ_{P} are diagonal in physical (gridpoint) space, where their diagonal elements are the standard deviations of background error for vorticity, divergence and geopotential; $\mathbf{W}_{R\zeta}$, $\mathbf{W}_{RP}$, $\mathbf{W}_{GD}$ and $\mathbf{W}_{GP}$ are diagonal in spectral space, and assign wavenumber-dependent proportions of the background error to balanced and unbalanced components; $\wp$ represents the operator which, when applied to a vorticity field, gives the corresponding linearly balanced geopotential.

The rationale behind this choice of change of variable is best understood by considering the implied inverse change of variable. Writing $\chi=(\chi_1,\chi_2,\chi_3)^{\mathrm{T}}$, the vorticity, divergence and linearized geopotential are given by

$$\begin{aligned} \zeta &= \mathbf{W}_{R\zeta}^{-\frac{1}{2}}\Sigma_{\zeta}\mathbf{C}^{\frac{1}{2}}\chi_1 \\ D &= \mathbf{W}_{GD}^{-\frac{1}{2}}\Sigma_{D}\mathbf{C}^{\frac{1}{2}}\chi_2 \\ P &= \wp(\zeta)+\mathbf{W}_{GP}^{-\frac{1}{2}}\left(\Sigma_{P}\mathbf{C}^{\frac{1}{2}}\chi_3-\mathbf{W}_{RP}^{\frac{1}{2}}\wp(\zeta)\right). \end{aligned} \tag{25}$$

For most wavenumbers, the diagonal elements of $\mathbf{W}_{RP}$ are zero. For these wavenumbers, the vectors χ_1, χ_2 and χ_3 represent normalized versions of the vorticity, divergence and the unbalanced part of the linearized geopotential. The balanced part of the geopotential is determined from the vorticity. This ensures that correlations between vorticity and the balanced component of geopotential are correctly represented. For a few low wavenumbers, corresponding to tidal modes, the diagonal elements of $\mathbf{W}_{RP}$ are set equal to the corresponding elements of $\mathbf{W}_{GP}$. This reduces to zero the correlations between mass and vorticity for tidal modes.

Eigenvectors of the Hessian of the cost function were calculated for an analysis for 12z on 1/6/1995. After 82 iterations of minimization, 52 eigenvectors were known to a relative accuracy of 10%. The analysis used the same observations as were used in the ECMWF operational analysis for the same date, with the exception of scatterometer measurements, which were removed since a linearized version of the observation operator for scatterometer data was not available. The standard deviations of background error currently used in the ECMWF variational assimilation have extremely large values over Antarctica. A consequence of this is that all the eigenvectors of the Hessian determined by the Lanczos algorithm have maximum amplitude over Antarctica. To avoid this problem, the standard deviations of background error for wind were set constant on model levels.

Figure 4 shows eigenvectors of the Hessian corresponding to the first, second, third and fifty-second largest converged eigenvalues. The vectors are plotted after transformation to physical space, and only the temperature component on model level 18 (close to 500hPa) is shown for each vector. The three leading eigenvectors have maximum amplitude over Europe, reflecting the high density of data in this region. Vectors corresponding to smaller eigenvalues show significant amplitudes at other geographical locations. The eigenvalues of the Hessian for the four vectors shown are approximately 1776, 750, 784 and 51.

Figures 5a and 5b show estimated differences between the standard deviations of background error and the standard deviations of analysis error for geopotential on model level 18. The estimate shown in figure 5b was produced using 52 eigenvectors, whereas the estimate shown in figure 5a was produced using the 24 eigenvectors which were known to 10% accuracy after 40 iterations of minimization. The two estimates have similar patterns, with strong maxima in the data-rich areas of Europe and North America, and weaker maxima over Alaska and eastern China. However, the maxima are much smaller when 24 vectors are used than when 52 vectors are used.

Equation 16 implies that an eigenvector has a significant influence on the approximate standard deviations of analysis error if the eigenvalue is not close to one and the eigenvector has significant amplitude when transformed to physical space. Both criteria are well satisfied by all the eigenvalues determined during the analysis. It is likely that a significant proportion of the true difference between the standard deviations of background and analysis error is unaccounted for by the known eigenvectors of the Hessian. Nevertheless, the similarity between the patterns of the differences between standard deviations of background and analysis error shown in fig. 5 suggests that it may be possible to account for the effect of unknown eigenvectors on the estimated standard deviations of analysis error by artificially increasing the influence of the known eigenvectors.

6. DIAGNOSING STANDARD DEVIATIONS OF BACKGROUND ERROR

In a multivariate analysis system, the imposition of balance conditions on the correlations of background error may result in standard deviations of background error which are not known explicitly. For example, the standard deviations of background error for linearized geopotential for the covariance matrix corresponding to the change of variable defined by equation 24 differ significantly from the standard deviations specified by Σ_p. It is useful, therefore, to be able to diagnose the standard deviations of background error. Two methods are discussed in this section.

If the background error covariance matrix is of the form $\mathbf{B}=\mathbf{L}\mathbf{L}^T$, the standard deviations of background error may be estimated using randomization. Consider a set of K random vectors $\chi_1,\chi_2,...,\chi_K$ drawn from a distribution with covariance matrix equal to the identity matrix. The elements of these vectors are independent random variables and are easy to produce using a random number generator. Applying $\mathbf{L}$ to each vector gives a set of vectors $\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_K$ drawn from a distribution with covariance matrix $\mathbf{B}$. The sample standard deviation of each element of $\mathbf{x}$ gives an estimate of the actual standard deviation implied by $\mathbf{B}$.

Each element of each random vector $\mathbf{x}_i$ is a weighted sum of elements of χ_i, the weights being the elements of one row of $\mathbf{L}$. The central limit theorem applies, so to a good approximation, the elements of $\mathbf{x}_i$ are normally distributed. The sample mean variance of each element of $\mathbf{x}$ is consequently a chi-squared variable. Elementary statistics shows that the standard deviation of the relative error in the estimated standard deviation of the element is $(2K)^{-1/2}$. Thus, 50 vectors are required to achieve a standard deviation of relative error for the estimated standard deviations of 10%.

An alternative method for estimating the standard deviations of background error is to approximate $\mathbf{B}$ by $\mathbf{V}\Lambda\mathbf{V}^T$, where $\mathbf{V}$ is a rectangular matrix consisting of a few leading eigenvectors of $\mathbf{B}$, and where Λ is a diagonal matrix whose diagonal entries are the corresponding eigenvalues. The eigenpairs may be determined using a Lanczos algorithm. The approximate standard deviations of background error are given by calculating the diagonal of $\mathbf{V}\Lambda\mathbf{V}^T$. An advantage of this approach is that it does not require that $\mathbf{B}$ is of the form $\mathbf{L}\mathbf{L}^T$.

Both approximations to $\mathbf{B}$ may be written in the form $\mathbf{W}\mathbf{W}^T$, where $\mathbf{W}$ is a rectangular matrix containing a few vectors. Replacing $\mathbf{W}$ with $\mathbf{S}\mathbf{W}$, where the matrix $\mathbf{S}$ represents a change of variable, gives an approximation to $\mathbf{S}\mathbf{B}\mathbf{S}^T$, the background error covariance matrix for the transformed variable $\psi=\mathbf{S}\mathbf{x}$. For example, if $\mathbf{S}$ represents projection onto Hough modes, the implied standard deviations of background error for each mode may be estimated.

Figure 6 shows standard deviations of background error for the simplified analysis system described in section 4, estimated using the two techniques. 50 random vectors were used for the randomization approximation and 25 eigenpairs were used for the eigenvector-based approximation. The eigenpairs were determined using 40 iterations of a Lanczos algorithm. The randomization approximation has large oscillations around the true standard deviations. However, the oscillations are of a smaller scale than the spatial variation of the true standard deviations of background error and could be removed by filtering to leave an accurate approximation to the standard deviations of background error. The eigenvector-based approximation is smoother than the randomization approximation, but underestimates the true standard deviations of background error. This is because the approximation effectively sets the eigenvalues for unknown eigenvectors to zero.

The randomization approximation of **B** has been implemented in the ECMWF variational analysis system for the change of variable defined by equation 24. Figure 7 shows an estimate, calculated from 50 gradients, of zonal mean standard deviations of background error for geopotential for the analysis discussed in section 5. The approximate standard deviations show many expected features. In particular, the standard deviations are smallest in the tropics in the lower troposphere and increase with altitude and distance from the equator, reaching maxima at high northern and southern latitudes in the upper troposphere. Note that the standard deviations are nearly symmetric about the equator, since for this analysis, the standard deviations for wind were set constant on model levels.

7. PRECONDITIONING

Preconditioning can significantly increase the speed of convergence of the conjugate gradient minimization algorithm. Consider the quadratic function

$$f(x)=\frac{1}{2}x^T\mathbf{M}x+b^Tx+c \tag{26}$$

The preconditioned conjugate gradient algorithm for minimizing f is equivalent to conjugate gradient minimization of

$$g(y)=\frac{1}{2}y^T\mathbf{U}^{-\frac{1}{2}}\mathbf{M}\mathbf{U}^{-\frac{1}{2}}y+(\mathbf{U}^{-\frac{1}{2}}b)^Ty+c \tag{27}$$

where $\mathbf{y}=\mathbf{U}^{1/2}\mathbf{x}$, and where $\mathbf{U}^{1/2}$ is the symmetric square root of a positive definite matrix $\mathbf{U}$, called the preconditioner. For fast convergence, the preconditioner should be chosen to approximate $\mathbf{M}$. In particular, convergence will be fast if the eigenvalues of $\mathbf{U}^{-1/2}\mathbf{M}\mathbf{U}^{-1/2}$ are "clustered".

In a cycling analysis system, it is probable that the Hessian matrix of J^o will vary little from day to day, since the locations of many of the observations and the covariances of observation error will be unchanged. This suggests that a preconditioner may be constructed from vectors generated at an earlier analysis cycle.

Suppose we have an approximation to the Hessian matrix for an earlier analysis cycle of the form

$$J''\approx\hat{\mathbf{B}}^{-1}+\hat{\mathbf{W}}\hat{\mathbf{W}}^T \tag{28}$$

If the background error covariance matrix changes little from cycle to cycle, then a good approximation to the Hessian for the current cycle is provided by $\mathbf{B}^{-1}+\hat{\mathbf{W}}\hat{\mathbf{W}}^T$. Applying the change of variable $\mathbf{L}$, where $\mathbf{B}=\mathbf{L}\mathbf{L}^T$, gives

$$\mathbf{L}^TJ_x''\mathbf{L}\approx\mathbf{I}+(\mathbf{L}^T\hat{\mathbf{W}})(\mathbf{L}^T\hat{\mathbf{W}})^T. \tag{29}$$

Now, $(\mathbf{L}^T\hat{\mathbf{W}})(\mathbf{L}^T\hat{\mathbf{W}})^T$ is a matrix of low rank, and so has only a few non-zero eigenvalues. The eigenpairs corresponding to the non-zero eigenvalues may be determined by applying a sequence of Householder reductions to $(\mathbf{L}^T\hat{\mathbf{W}})$ to reduce it to upper triangular form. This is equivalent to performing an orthonormal

(eigenvalue preserving) transformation of $(\mathbf{L}^T\hat{\mathbf{W}})(\mathbf{L}^T\hat{\mathbf{W}})^T$ to give a matrix which is zero everywhere except for a small block in the upper left corner. The eigenvalues of this block are the same as the non-zero eigenvalues of $(\mathbf{L}^T\hat{\mathbf{W}})(\mathbf{L}^T\hat{\mathbf{W}})^T$, and are easily found using standard methods. The corresponding eigenvectors of $(\mathbf{L}^T\hat{\mathbf{W}})(\mathbf{L}^T\hat{\mathbf{W}})^T$ are also easily determined from those of the small block.

Once the eigenpairs, $(\mu_k,\mathbf{v}_k)$ of $(\mathbf{L}^T\hat{\mathbf{W}})(\mathbf{L}^T\hat{\mathbf{W}})^T$ are known, a preconditioner is given by selecting eigenpairs for which the eigenvalue is close to one to give

$$\mathbf{U}=\mathbf{I}+\sum_{k=1}^{K}(\lambda_k-1)\mathbf{v}_k\mathbf{v}_k^T \tag{30}$$

where $\lambda_k=1/(1-\mu_k)$.

Figure 8 shows the effect on the convergence of the cost function of the ECMWF 3d-Var analysis system of preconditioning the combined conjugate gradient minimization with either 19 or 68 eigenpairs of the Hessian. In this case, the analysis for which the eigenpairs were calculated and the preconditioned analysis were for the same date. Consequently, the vectors from which the preconditioner was constructed are exact eigenvectors of the Hessian. The use of preconditioning clearly has a beneficial effect on the convergence rate for the first 20 iterations.

Also shown in fig. 8 is the convergence rate for the quasi-Newton method. The difference in convergence between the quasi-Newton method and the unpreconditioned conjugate gradient method is due the use of exact line searches in the latter. For a quadratic cost function, it is possible to implement exact line searches in the quasi-Newton algorithm. The convergence rates and computational costs of the quasi-Newton and (unpreconditioned) conjugate-gradient algorithms would then be equal. It should also be noted that the preconditioner described above can be used to accelerate the quasi-Newton algorithm.

8. TEMPORAL PROPAGATION

As discussed in section 2, one possibility for estimating the covariance matrix of short-term prediction error is to determine eigenpairs of **P**. In variational assimilation, this requires either the (computationally expensive) solution of the generalized eigenvalue problem represented by equation 3, or the solution of the ordinary eigenvalue problem $\mathbf{MAM}^T\mathbf{x}=\lambda\mathbf{x}$ in which the analysis error covariance matrix is approximated.

An alternative approach to the calculation of eigenvectors of **P** is to propagate directly an approximation to **A**. The approximate analysis error covariance matrices presented in this paper all take the form of low-rank corrections to **B**. In section 6, low-rank approximations to **B** were discussed. Combining the approximations for **A** and **B** gives approximations which may be propagated in time at the cost of transforming a few vectors. The approximations generated in this way will be less efficient than approximations based on the eigenvectors of **P**, but may be computationally cheaper to calculate, since they do not require the solution of an eigenvalue problem.

The randomization and Lanczos approximations for **A** may both be written in the form

$$\mathbf{A} \approx \mathbf{B} - \mathbf{V}\mathbf{V}^T \tag{31}$$

where $\mathbf{V}$ is a rectangular matrix consisting of a few vectors. Using one of the approximations to $\mathbf{B}$ described in section 6, we may write $\mathbf{B} \approx \mathbf{W}\mathbf{W}^T$. This gives a (rank-deficient) approximation to $\mathbf{A}$ which is easily propagated in time to give

$$\mathbf{P} \approx (\mathbf{M}\mathbf{W})(\mathbf{M}\mathbf{W})^T - (\mathbf{M}\mathbf{V})(\mathbf{M}\mathbf{V})^T. \tag{32}$$

This approximation has been calculated for the simplified analysis system described in section 4. In this case, the model for which $\mathbf{M}$ is the tangent linear operator integrates Burger's equation

$$\frac{\partial x}{\partial t} + x\frac{\partial x}{\partial s} - \mu\frac{\partial^2 x}{\partial s^2} = 0 \tag{33}$$

where s represents distance in metres around the 45° constant-latitude circle. The period of the domain, L, is roughly 28.3×10^6m and the diffusion coefficient, μ, is $10^5 m^2 s^{-1}$. At initial time, the true state is defined as $x = 20 + 15\sin(2\pi s/L)$. The equation is discretized by representing x as a truncated fourier series. The nonlinear term is calculated using the transform method on a non-aliasing grid and Euler-backward (Matsuno) time integration is used. Figure 9 shows the true initial state and the background state for the analysis, together with 12 hour and 24 hour forecasts from the background state

Standard deviations of 24 hour forecast error for the simplified system were estimated using equation 32. $\mathbf{W}$ was constructed from 25 eigenvectors of $\mathbf{B}$ and $\mathbf{V}$ was constructed from 31 eigenvectors of the Hessian. The estimated standard deviations of forecast error are shown in figure 10. Also shown are the standard deviations of forecast error corresponding to the approximation $\mathbf{U}\Lambda\mathbf{U}^T$, where $\mathbf{U}$ contains 28 eigenvectors of the approximate forecast error covariance matrix $\mathbf{M}\hat{\mathbf{A}}\mathbf{M}^T$ in which $\hat{\mathbf{A}}$ is the Lanczos approximation to the analysis error covariance matrix based on 35 eigenvectors. Both methods capture the main features of the true standard deviations of forecast error. However, the use of an approximate covariance matrix of background error for the direct propagation method gives an approximation to $\mathbf{P}$ which is not positive definite. This gives negative variances of forecast error, indicated in figure 10 by the regions where the approximate standard deviations drop below zero. The direct propagation method also uses twice as many vectors as the eigenvector-based approximation. However, the number of integrations of the tangent linear model required to directly propagate the approximate covariance matrix of analysis error is equal to the number of vectors, whereas the eigenvector-based approximation requires one integration of the tangent linear model and one integration of the adjoint model for each iteration of the Lanczos algorithm. The 28 eigenpairs used for the eigenvector-based approximation were determined after 50 iterations. Thus, direct propagation of the approximate analysis error covariance matrix required fewer integrations.

Direct propagation of $\mathbf{A}$ using the randomization approximation to $\mathbf{B}$ was also tried. The approximate standard deviations of forecast errors (not shown) had large spurious oscillations which could only be reduced to reasonable levels by using impractically large numbers of random vectors.

The BFGS approximation to $\mathbf{A}^{-1}$, and the corresponding approximation to $\mathbf{A}$ are of a less simple form than the Lanczos and randomization approximations. Nevertheless, the transformation property of the BFGS

update formula allows a simple expression of an approximation to the covariance matrix of forecast error. Replacing **U** by **M** in equation 19 gives

$$\mathbf{P} \sim F(\mathbf{M}\mathbf{G}^{(k-1)}\mathbf{M}^{*}, (\mathbf{M}^{*})^{-1}\mathbf{y}_k, \mathbf{M}\mathbf{s}_k). \tag{34}$$

The vectors $\mathbf{M}\mathbf{s}_k$ are differences in **x** propagated by the tangent linear model. In a 4d-Var analysis, these vectors are generated during the minimization. The vectors $\mathbf{y}_k$ are differences in gradients of the cost function evaluated at the initial time, t_0, of the forecast. In general, the operator $(\mathbf{M}^{*})^{-1}$ is not available, so it is not possible to calculate the BFGS approximation to the forecast error covariance matrix. However, in a 4d-Var analysis for which observations occur only at or after the time, t_1, at which the approximate forecast error covariance matrix is required, $\mathbf{M}^{*}$ propagates the gradient from time t_1 to time t_0. In this case, the vectors $(\mathbf{M}^{*})^{-1}\mathbf{y}_k$ are differences in ∇J evaluated at t_1. These vectors are generated during the 4d-Var analysis. Thus, for the restricted circumstances of a 4d-Var analysis with observations only at or after t_1, an approximation to **P** may be calculated from information generated during the minimization of the cost function.

It is well known (*Ghil and Malanotte-Rizzoli*, 1991) that, for linear dynamics and negligible model error, 4d-Var and the Kalman filter produce identical analyses for the end of the 4d-Var analysis period. In a Kalman filter, an analysis step is performed at the validity time of each observation. The period between observations is usually small, so the approximations of linear dynamics and negligible model error hold to good accuracy, and the analysis step of the Kalman filter may be performed using a 4d-Var algorithm. This 4d-Var analysis uses observations only at the end of the analysis period. Consequently, the BFGS approximation to the forecast error covariance matrix may be generated from the vectors calculated during minimization of the cost function. Thus, by replacing the true forecast error covariance matrix of the Kalman filter by the BFGS approximation, an efficient approximate Kalman filter may be implemented. The effects of model error may be incorporated by modifying the approximate forecast error covariance matrix by, for example, adding a matrix representing the covariance matrix of model error.

9. CONCLUSIONS

Three methods for estimating the covariance matrix of analysis error in incremental variational data assimilation have been presented. The methods are sufficiently computationally efficient to be used in an operational analysis system. All three methods produce reasonable estimates of the variance and covariance structure of analysis error in a one-dimensional, univariate implementation of 3d-Var.

The most promising method (the Lanczos algorithm) has been evaluated in a full meteorological 3d-Var analysis system. It has been demonstrated that the horizontal pattern of analysis error variance can be determined with relatively few vectors. The method also provides a good first-guess and a preconditioner which can be used to accelerate a subsequent analysis. However, a significant proportion of the difference between the variances of background and analysis error was not explained by the first 52 eigenvectors to be determined by the Lanczos algorithm.

Two low-rank approximations to the background error covariance matrix were demonstrated. The approximations allow diagnosis of the true standard deviations of background error for a multivariate analysis system in which the standard deviations of background error are not explicitly known.

The approximations to the covariance matrix of analysis error presented in this paper all consist of low-rank corrections applied to the covariance matrix of background error. Propagation of the approximations may be achieved by propagating the background error covariance matrix and the corrections. The former may be achieved with a few integrations of the tangent linear model if the background error covariance matrix is replaced by a low-rank approximation. For the randomization and Lanczos approximations, propagation of the corrections to the background error covariance matrix may also be achieved by a few integrations of the tangent linear model. For the BFGS approximation, propagation requires the inverse of the adjoint model. This is not generally available in 4d-Var. However, in the restricted case of a 4d-Var analysis with observations only at or after the time at which the forecast error covariance matrix is required, the information required to propagate the BFGS matrix is generated during minimization of the cost function. This allows an approximate Kalman filter to be implemented which is as computationally efficient as 4d-Var.

An alternative to direct propagation of an approximate covariance matrix of analysis error, is to approximate the forecast error covariance matrix using a few leading eigenvectors. These vectors may be calculated without approximation by solving a generalized eigenvalue problem. Unfortunately, existing algorithms for solving such problems are computationally expensive. A more efficient alternative is to solve the ordinary eigenvalue problem $\mathbf{MAM}^{\mathrm{T}}\mathbf{x}=\lambda\mathbf{x}$, in which $\mathbf{A}$ is approximated. This approach has advantages over the direct propagation method. In particular, the approximate forecast error covariance matrix is positive definite. The eigenvector-based approximation is also more efficient, requiring fewer vectors to be stored for comparable accuracy.

REFERENCES

Broyden C.G, 1969. A new double-rank minimization algorithm. Notices American Math. Soc., *16*, p 670.

Cohn S.E. and R. Todling, 1995. Approximate Kalman filters for unstable dynamics. Proc. International Symposium on Assimilation of Observations in Meteororogy and Oceanography, Tokyo, Japan, 13-17 March 1995.

Courtier, P., E. Andersson, W. Heckley, G. Kelly, J. Pailleux, F. Rabier, J-N. Thépaut, P. Unden, D. Vasiljevic, C. Cardinali, J. Eyre, M. Hamrud, J. Haseler, A. Hollingsworth, A. McNally, A. Stoffelen, 1993. Variational Assimilation at ECMWF. ECMWF Research Dept. Tech. Memo. 194.

Dennis J.E. jr. and J.J. Moré, 1977. Quasi-Newton methods, motivation and theory. SIAM review, *19*, 46-89.

Dharssi, I., A.C. Lorenc and N.B. Ingleby, 1992. Treatment of gross errors using maximum probability theory. Quart. J. Roy. Met. Soc., *118*, 1017-1036.

Ehrendorfer M. and J.J. Tribbia, 1995. Efficient prediction of covariances using singular vectors. Proc. 6th International Meeting on Statistical Climatology, Galway, Ireland, 19-23 June 1995.

Gilbert J.Ch. and C. Lemaréchal, 1989. Some numerical experiments with variable storage quasi-Newton algorithms. Math. Programming, *45*, 407-435.

Ghil M. and P. Malanotte-Rizzoli, 1991. Data assimilation in meteorology and oceanography. Adv. Geophys., *33*, 141-266.

Ide K., A.F. Bennett, P. Courtier, M. Ghil, A.C.Lorenc, 1995. Unified Notation for Data Assimilation: Operational, Sequential and Variational. J. Met. Soc. Japan, (to appear).

Kalman R., 1960. A new approach to linear filtering and prediction problems. Trans. ASME, Ser.D, J. Basic Eng., *82*, 35-45.

Lanczos C, 1950. An iteration method for the solution of the eigenvalue problem of linear differential and integral operators. J. Res. Nat. Bur. Standards, *45*, 255-282.

Nocedal J, 1980. Updating quasi-Newton matrices with limited storage. Mathematics of Computing, *35*, No. 151, 773-782.

Paige C.C. and M.A. Saunders, 1975. Solution of sparse indefinite systems of linear equations. SIAM J. Numerical Analysis, *12*, 617-629.

Rabier F. and P. Courtier, 1992. Four-dimensional assimilation in the presence of baroclinic instability. Quart. J. Roy. Met. Soc., *118*, 649-672.

Rabier F., P. Courtier, J. Pailleux, O. Talagrand, J-N. Thépaut and D. Vasiljevic, 1992. Comparison of four-dimensional variational assimilation with simplified sequential assimilation. Proc. workshop on variational assimilation with special emphasis on three dimensional aspects, 9-12 November 1992.

Scott D.S., 1981. Solving sparse symmetric generalized eigenvalue problems without factorization. SIAM J. Numerical Analysis, *18*, 102-110.

Sherman J. and W.J. Morrison, 1949. Adjustment of an inverse matrix corresponding to changes in the elements of a given column or a given row of the original matrix. Ann. Math. Statist., *20*, 621.

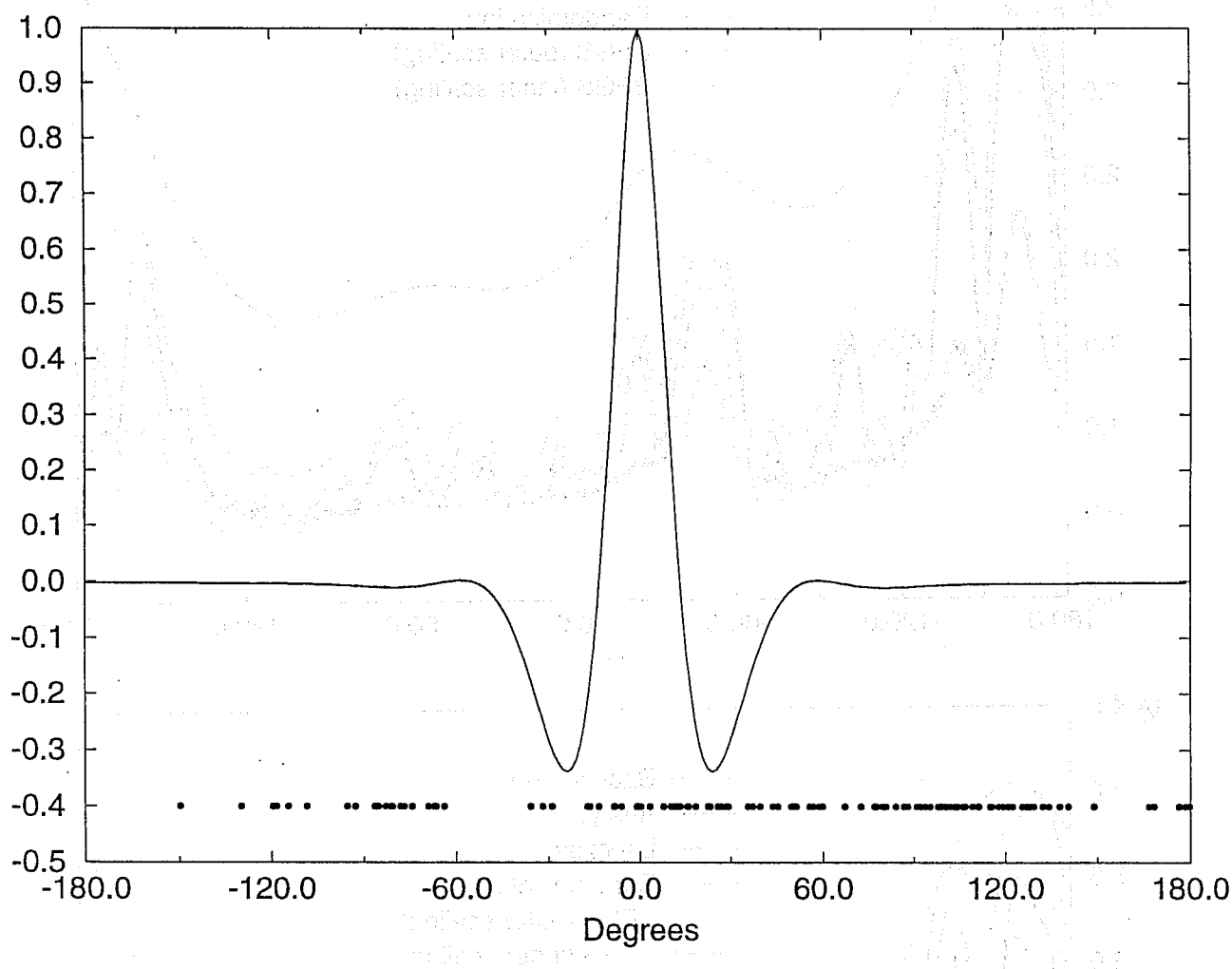

Fig. 1: Correlation structure function and observation locations for the simplified analysis system. The locations of the observations are shown by dots.

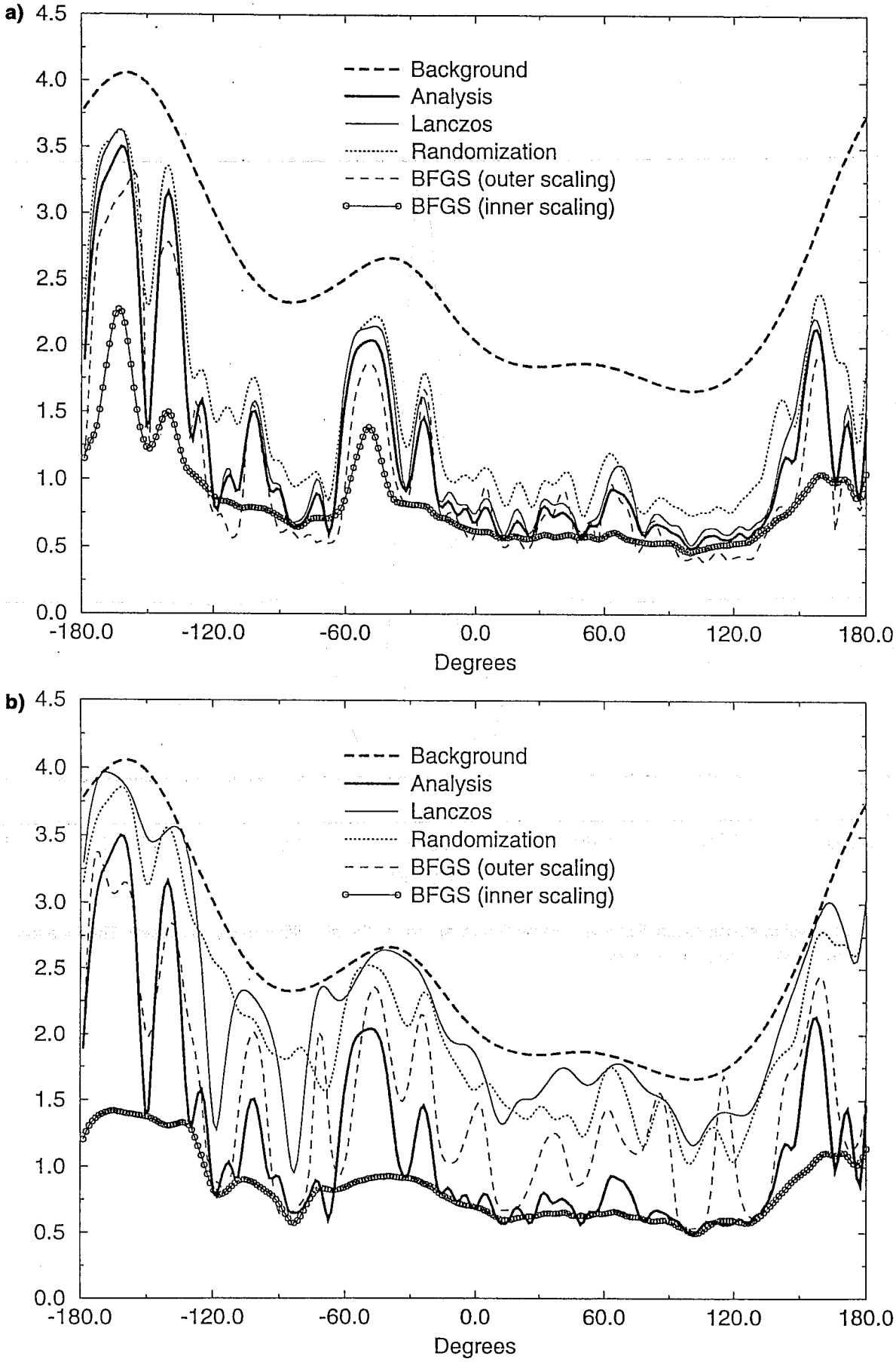

Fig. 2: Estimated standard deviations of analysis error for the simplified analysis system produced (a) using 35 gradients, (b) using 10 gradients. The standard deviations of background error and the true standard deviations of analysis error are also shown.

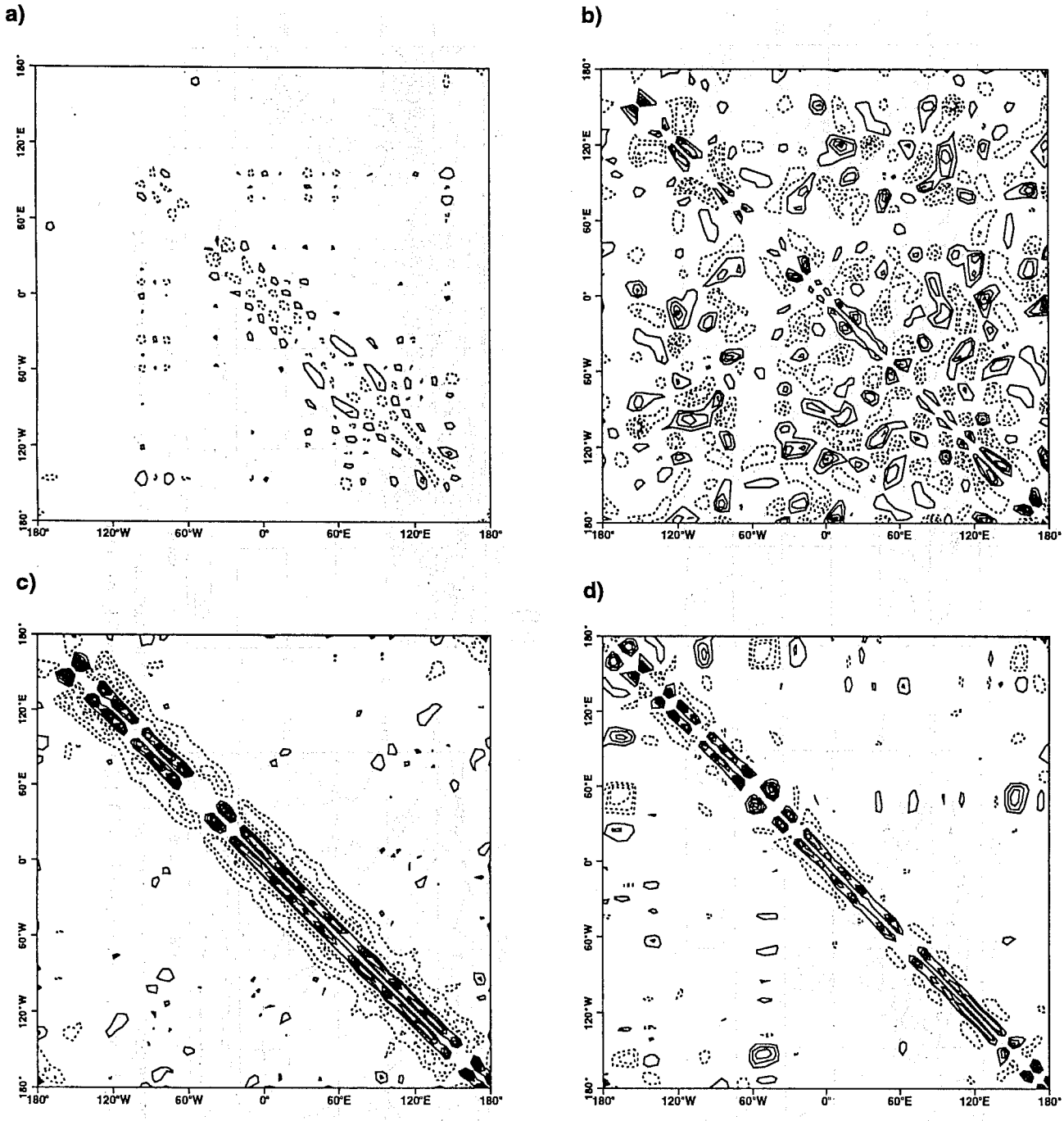

Fig. 3: Differences between approximate and true correlation matrices: (a) Lanczos method, (b) Randomization, (c) BFGS method with outer scaling, (d) BFGS method with inner scaling.

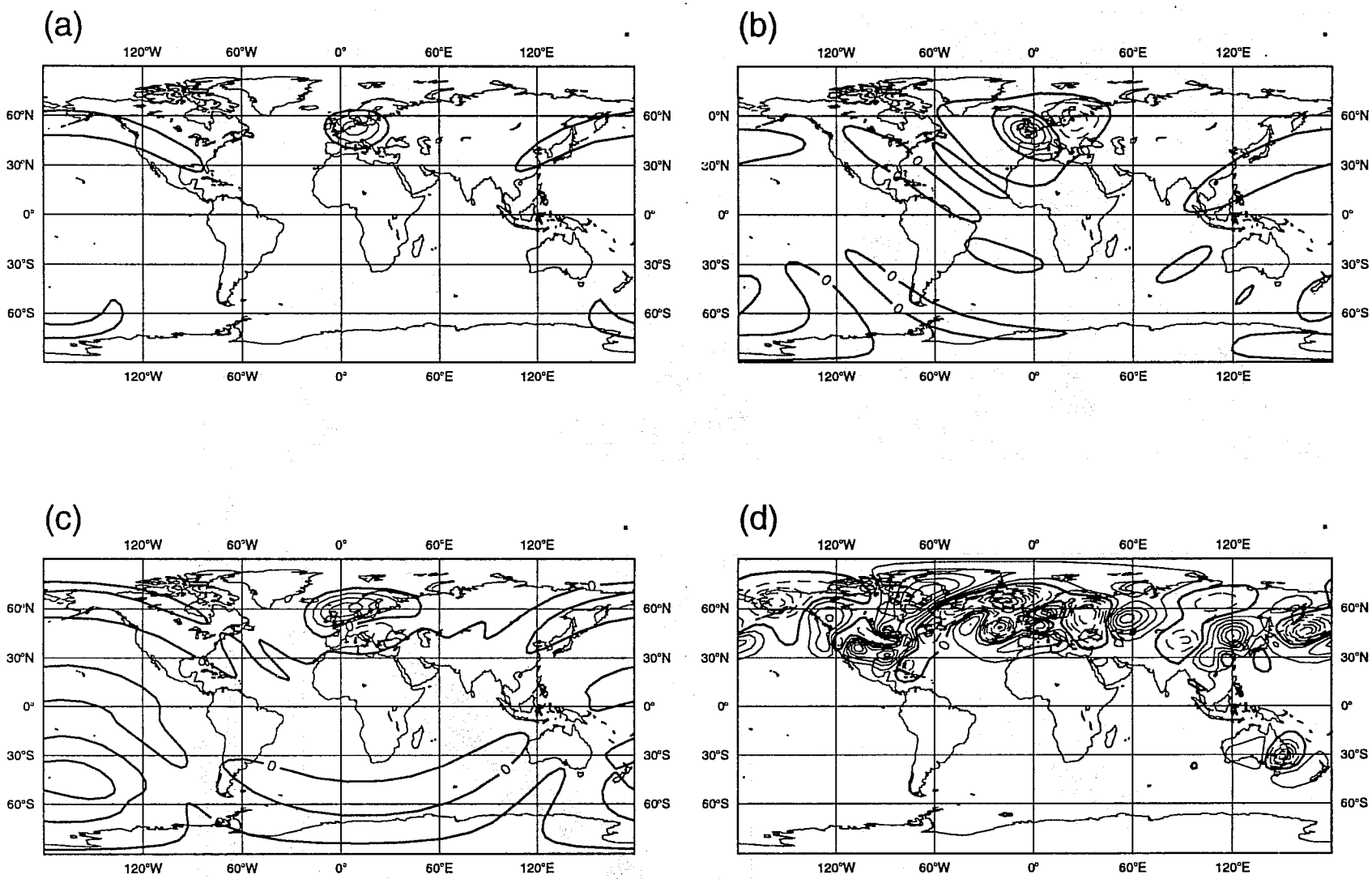

Fig. 4: The temperature field on model level 18 for four eigenvectors of the Hessian of the cost function for a 3d-Var analysis for 12z 1/6/1995. (a) λ=1776, (b) λ=750, (c) λ=784, (d) λ=51.

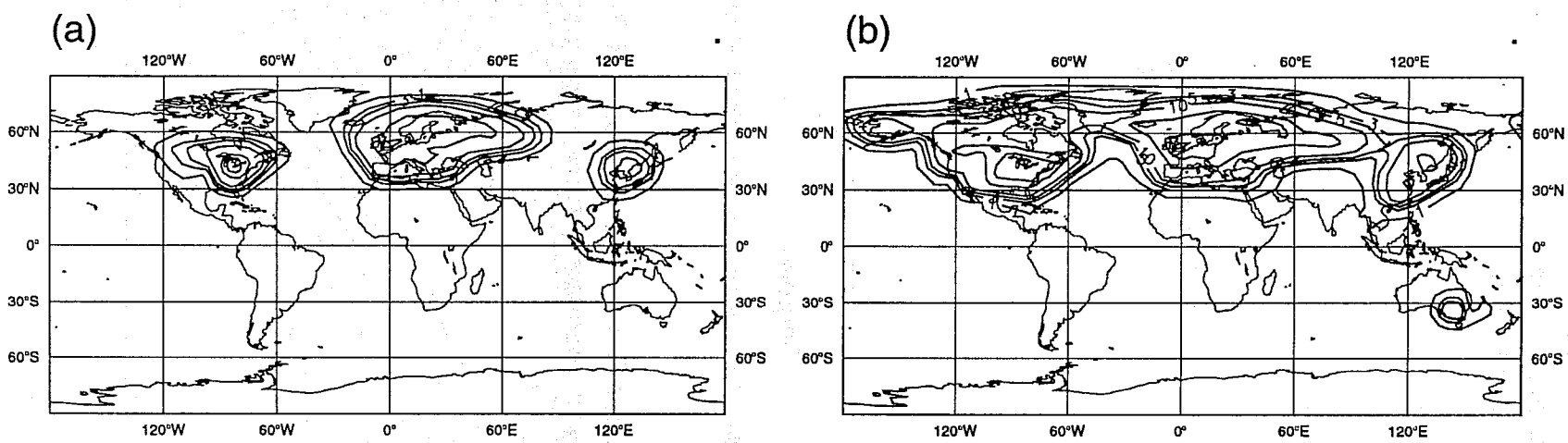

Fig. 5: Differences between the standard deviations of background error and analysis error for geopotential on model level 18, estimated using (a) 52 vectors, (b) 24 vectors. Contours are shown for differences of differences of 1, 3, 5, 10, 30, 50 and 100 m^2s^{-2}.

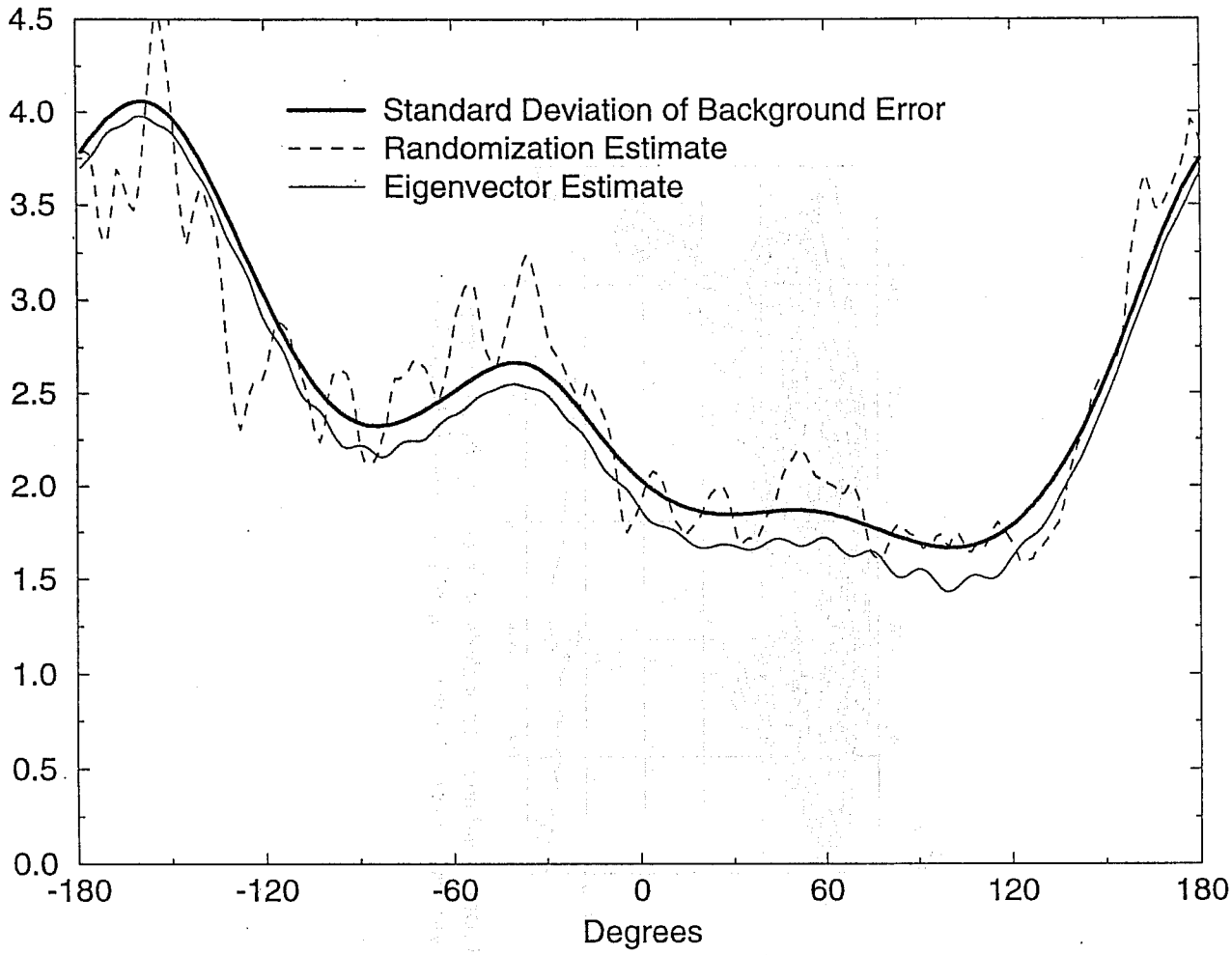

Fig. 6: Two estimates of the standard deviation of background error for the simplified analysis system.

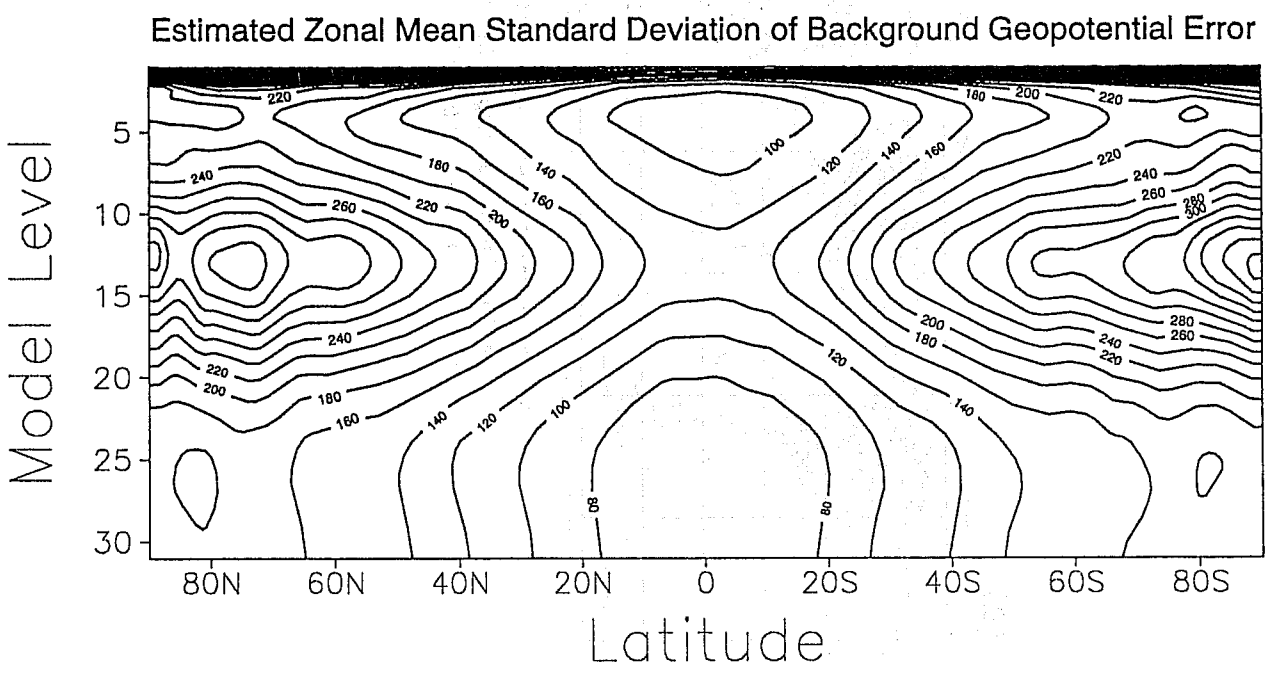

Fig. 7: Zonal mean standard deviation of background error for an analysis for 12z 1/6/1995, estimated using randomization.

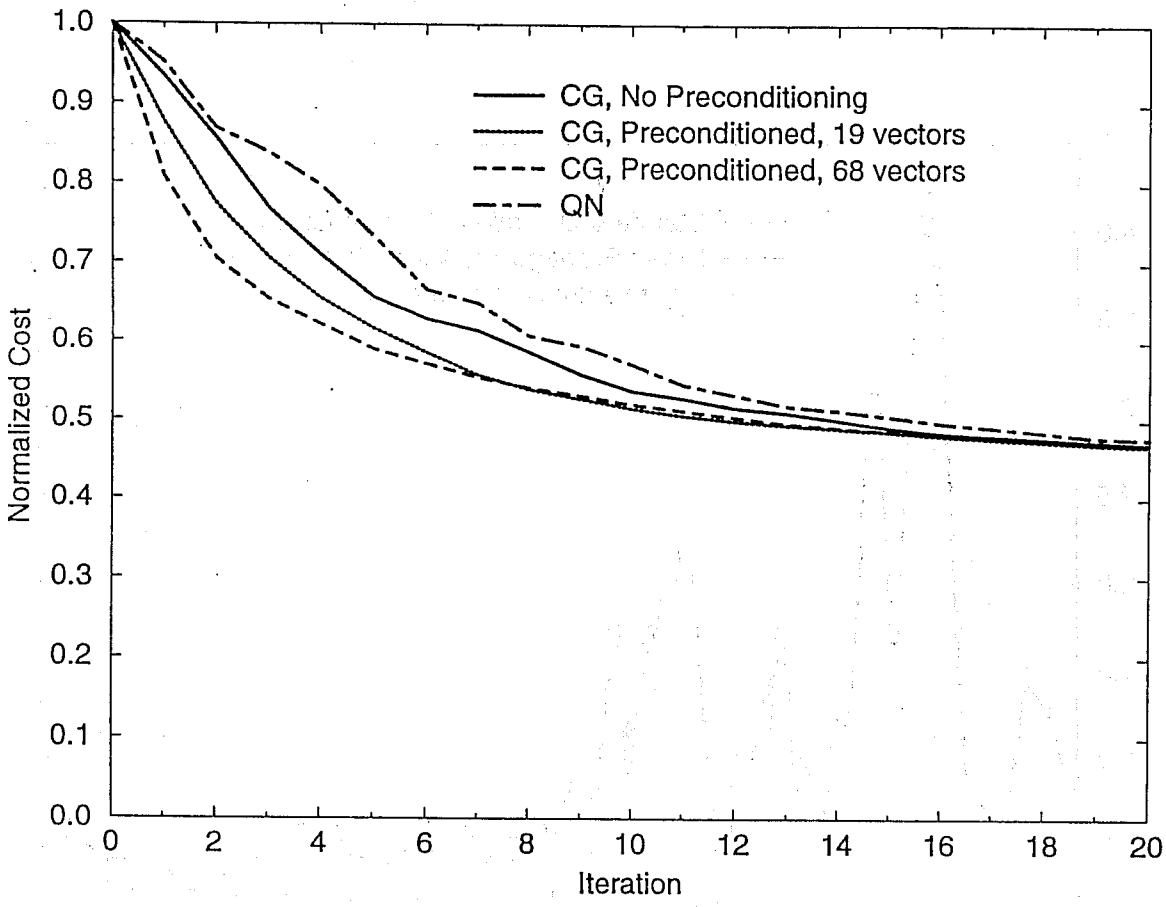

Fig. 8: Normalized cost as a function of iteration for the conjugate gradient algorithm with varous amounts of preconditioning, and for a quasi-Newton algorithm.

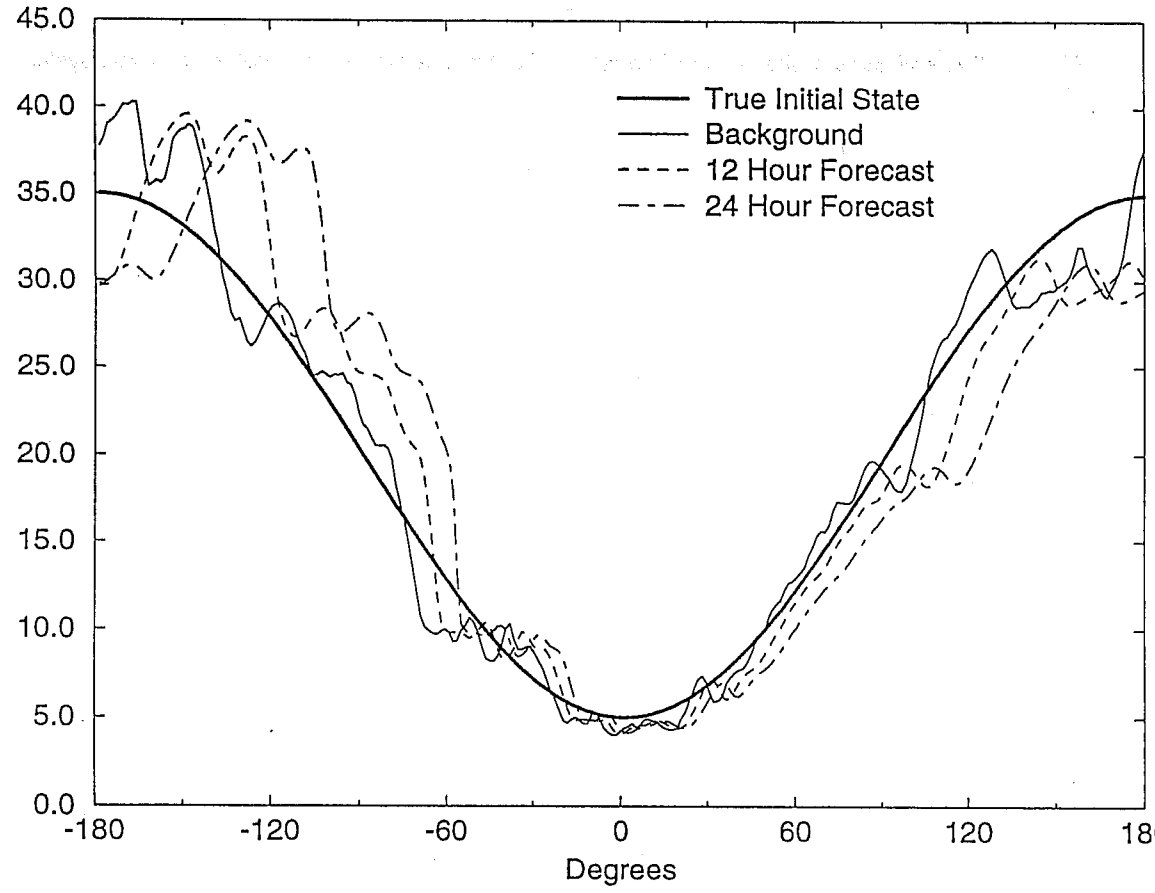

Fig. 9: True initial state and background state for the simplified analysis system, together with 12 hour and 24 hour forecasts from the background state.

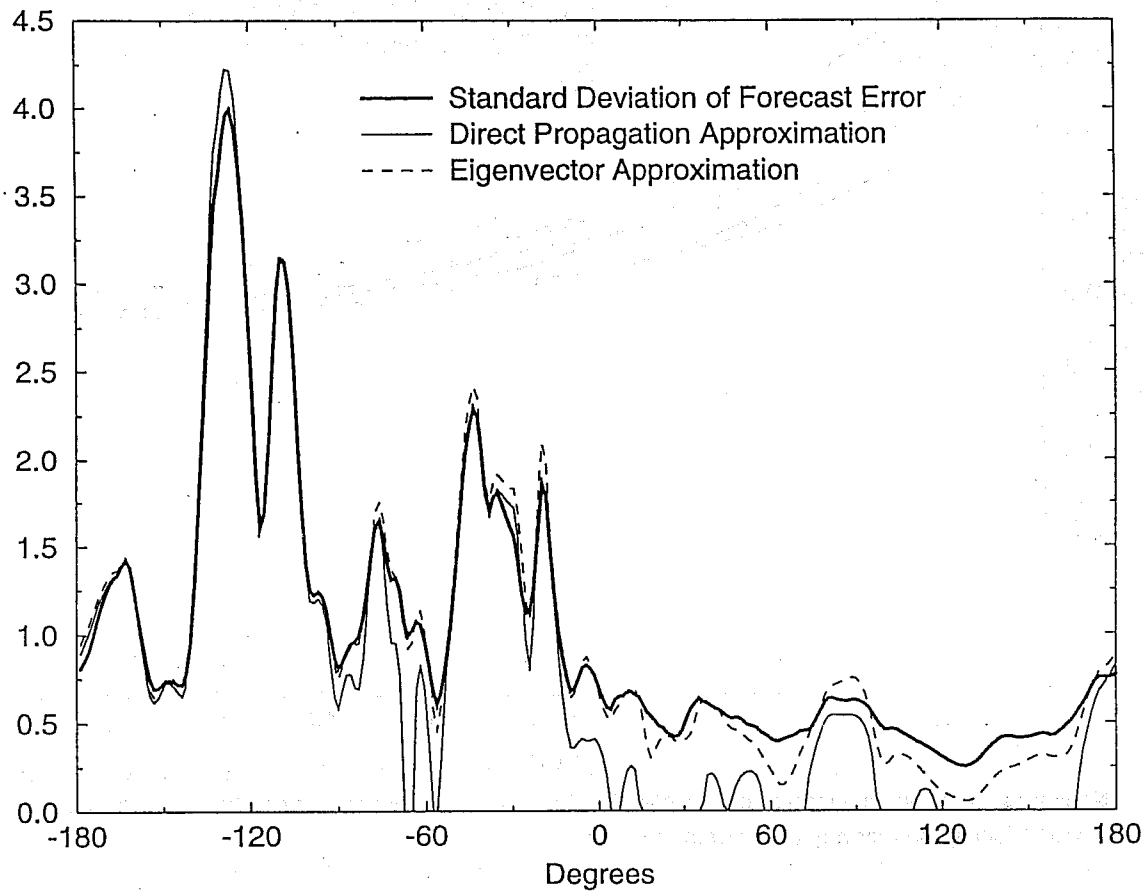

Fig. 10: Two estimates of the standard deviation of forecast error for the simplified analysis system.

Appendix A: **The Combined Conjugate Gradient and Lanczos Algorithm**

Consider the quadratic function

$$f(x) = \frac{1}{2} x^T M x + b^T x + c. \tag{35}$$

Given an initial point, $\mathbf{x}_0$, and an initial descent direction, $\mathbf{d}_0$, the conjugate gradient algorithm for minimizing f is

$$\begin{aligned} x_{k+1} &= x_k + \alpha_k d_k \\ g_{k+1} &= -\nabla f(x_{k+1}) \\ d_{k+1} &= g_{k+1} + \gamma_k d_k \end{aligned} \tag{36}$$

where $\gamma_k = (g^T_{k+1} g_{k+1})/(g^T_k g_k)$ and where α_k is the value of α which minimizes f along $\mathbf{x}_k + \alpha \mathbf{d}_k$.

By eliminating $\mathbf{d}_k$ and $\mathbf{d}_{k+1}$ from equations 36 and noting that $\mathbf{M}(\mathbf{x}_{k+1} - \mathbf{x}_k) = (\mathbf{g}_{k+1} - \mathbf{g}_k)$, it is straightforward to show that $\mathbf{g}_k$ satisfies the following three-term recurrence

$$\frac{1}{\alpha_{k+1}} g_{k+2} + (M - \delta_{k+1} I) g_{k+1} + \frac{\gamma_k}{\alpha_k} g_k = 0 \tag{37}$$

where $\delta_{k+1} = (1/\alpha_{k+1}) + (\gamma_k/\alpha_k)$.

Now define $\mathbf{q}_k = c_k \mathbf{g}_k$ where $c_k = (g^T_k g_k)^{-1/2}$. The vectors $\mathbf{q}_k$ are orthonormal and satisfy

$$M q_k = \beta_{k+1} q_{k+1} + \delta_k q_k + \beta_k q_{k-1} \tag{38}$$

where $\beta_{k+1} = -c_k/(\alpha_k c_{k+1})$.

This is precisely the recurrence used by the Lanczos algorithm to determine eigenpairs of **M**.

Appendix B: **Proof of identities 1 and 2**

Let $\mathbf{a},\mathbf{b},\mathbf{x} \in \mathbb{R}^n$, $\mathbf{A} \in \mathbb{R}^n \times \mathbb{R}^n$, and let $\langle \cdot,\cdot \rangle$ and $\times$ be respectively an inner product and the corresponding tensor product defined on $\mathbb{R}^n$.

Consider $[(\mathbf{a}\times\mathbf{b})\mathbf{A}]\mathbf{x}$. We have

$$\begin{aligned}
[(a\times b)\mathbf{A}]x &= (a\times b)(\mathbf{A}x) \\
&= \langle b,\mathbf{A}x\rangle a \\
&= \langle \mathbf{A}^* b,x\rangle a \\
&= [a\times(\mathbf{A}^* b)]x.
\end{aligned}$$

Since this holds for arbitrary $\mathbf{x}$, we must have $(\mathbf{a}\times\mathbf{b})\mathbf{A} \equiv \mathbf{a}\times(\mathbf{A}^*\mathbf{b})$. This proves identity 1.

Next, consider $\langle(\mathbf{a}\times\mathbf{b})\mathbf{x},\mathbf{y}\rangle$. We have

$$\begin{aligned}
\langle(a\times b)x,y\rangle &= \langle\langle b,x\rangle a,y\rangle \\
&= \langle b,x\rangle\langle a,y\rangle \\
&= \langle a,y\rangle\langle b,x\rangle \\
&= \langle(b\times a)y,x\rangle \\
&= \langle(b\times a)^* x,y\rangle.
\end{aligned}$$

This holds for arbitrary $\mathbf{x}$ and $\mathbf{y}$, giving $(\mathbf{a}\times\mathbf{b})^* \equiv (\mathbf{b}\times\mathbf{a})$. Identity 2 is now easily proved using this result and identity 1:

$$\begin{aligned}
\mathbf{A}(a\times b) &= ((a\times b)^*\mathbf{A}^*)^* \\
&= ((b\times a)\mathbf{A}^*)^* \\
&= ((b\times(\mathbf{A}a))^* \\
&= (\mathbf{A}a)\times b.
\end{aligned}$$