Tellus (1993), 45A, 478–492
Printed in Belgium – all rights reserved

TELLUS
ISSN 0280–6495

A preconditioning algorithm for large-scale minimization problems

By MILIJA ŽUPANSKI, *UCAR Visitor Research Program, NOAA/National Meteorological Center, Washington, DC 20233, USA*

(Manuscript received 2 November 1992; in final form 21 June 1993)

ABSTRACT

A new preconditioning algorithm is proposed, employing a Taylor series expansion of the cost-function, and the relation between the adjustment of the control variable and the computed gradient norm. The preconditioning matrix is a positive definite diagonal matrix, being a product of two positive definite diagonal matrices. One is the weight matrix related to the Hessian matrix definition in the case of identity model operator ("rough" scaling), and the other matrix is interpreted as a refined scaling of the control variable. The procedure is quite easy to implement, and the computer time and space requirements are negligible. The algorithm was tested in two cases of realistic four-dimensional variational data assimilation experiments, performed using an adiabatic version of the NMC's new regional forecast model and operationally obtained optimal interpolation analyses. Test results show a significant improvement in the decrease of the cost-function and the gradient norm when using the new preconditioning procedure. The preconditioning was applied to a memoryless quasi-Newton method, however the technique should be applicable to other minimization algorithms.

1. Introduction

Recent four-dimensional (4-D) variational data assimilation experiments (e.g., Thepaut and Courtier, 1991; Navon et al., 1992; Chao and Chang, 1992) have indicated a potential for the practical use of the adjoint method. More recent experiments which directly assimilated observations (Thepaut et al., 1993), have shown that the forecast with assimilated data performed better than the forecast starting from conventionally analyzed initial conditions. Experiments performed by M. Zupanski (1993) (hereafter Z93), showed that the assimilation of analyses is actually possible using a simplified adjoint while performing an assimilation using full-physics operational version of the forecast model. Forecasts with assimilated data performed noticeably better than the forecast starting from the conventional optimal interpolation (OI) analysis. However, an operational implementation of the method is still computationally prohibitive. A necessary requirement for a practical 4-D variational data assimilation is fast convergence of the minimization algorithm. Quasi-Newton methods (Luenberger, 1984; Gill et al., 1981) have proved useful in applications to the 4-D data assimilation (Navon and Legler, 1987). The inverse of an approximation to the Hessian matrix (second derivative of the cost-function with respect to the control variable) is created during minimization, making very efficient descent direction determination. In practice, only information from the first few iterations can be saved due to large memory requirements. Since problems in meteorology and oceanography contain many degrees of freedom, conjugate gradient methods (e.g., Luenberger, 1984) also have some advantage due to their low memory requirements and potentially good convergence rate. However, a conjugate direction property can be easily destroyed due to round-off errors and inaccurate line search (step-length determination). The method of Shanno (1978), a memoryless quasi-Newton method, incorporates some of the best qualities of both the quasi-Newton and conjugate gradient method, and it has been suggested as a good choice

in 4-D data assimilation problems (Navon and Legler, 1987). This method may be viewed as a conjugate gradient method with inaccurate line search. If the line search is exact, the method reduces to the Polak-Ribiere conjugate gradient method (Shanno, 1978).

Even with good general properties of these minimization methods, the rate of convergence may be very slow. Convergence properties of the minimization are determined by the Hessian matrix of the problem (Luenberger, 1984; Gill et al., 1981), with the speed of convergence related to the condition number of the Hessian (the ratio between its maximum and minimum eigenvalue). Geometrically, isopleths of the cost-function form a hyper-ellipsoid with the ratio between squares of the dominant (maximum and minimum) axes given by the condition number of the Hessian. The larger the condition number, the larger the stretching of the ellipsoid. In the case of large distortion of isopleths, i.e., when the Hessian matrix is ill-conditioned, the calculated descent direction may be almost orthogonal to the optimal direction. This may cause an extremely slow convergence, and even a divergence in some cases.

A way to speed-up the convergence and to "relax" ill-conditioning of the Hessian is to introduce a preconditioning matrix, $\boldsymbol{P}$, which will make the ellipsoid-like shape of the cost-function more spherical. The relevant condition number is now of the matrix $\boldsymbol{P}^{-1}\boldsymbol{H}$ ($\boldsymbol{H}$ denotes the Hessian matrix) and for fast convergence would be close to one (e.g., Luenberger, 1984). Then, the direction of the negative gradient would lead almost directly to the minimum. A hypothetical good preconditioner (e.g., Axelsson and Barker, 1984): (1) should be positive definite, (2) can be factored as $\boldsymbol{P}=\boldsymbol{E}\boldsymbol{E}^{\mathrm{T}}$ ($\boldsymbol{E}$ is an invertible matrix, and superscript T denotes a conjugate transpose, i.e., adjoint), (3) spectral condition number $K(\boldsymbol{H}_{\mathrm{p}})$ is much smaller than $K(\boldsymbol{H})$ for solving $\boldsymbol{H}\boldsymbol{x}=\boldsymbol{b}$, where $\boldsymbol{H}_{\mathrm{p}}=\boldsymbol{E}^{-1}\boldsymbol{H}\boldsymbol{E}^{-\mathrm{T}}$ is a preconditioned Hessian matrix, (4) the preconditioned system should be solved more efficiently than $\boldsymbol{H}\boldsymbol{x}=\boldsymbol{b}$, and (5) computational time and space requirements should be relatively small in relation to $\boldsymbol{H}$.

The problem of preconditioning has not received enough attention in 4-D variational data assimilation minimization algorithms. Some aspects are generally addressed by Thacker (1989). Unfortunately, large scale minimization problems do not allow explicit treatment of the Hessian matrix due to high computer storage and time requirements. There is no efficient general preconditioner applicable to all problems, and one has to make an effort to develop a preconditioner suitable for data assimilation problems. Related to practical applications, one should try to satisfy another property of a good preconditioner, a "monotonic" convergence. "Monotonic" is here used in the sense that both the cost-function and the gradient norm should decrease in each iteration. Our experience with the algorithm of Shanno (1978) is that monotonic decrease of the gradient norm is very important property, and, if not satisfied, it usually implies a problem in minimization. In general, the monotonic decrease of the gradient norm might not be a necessary requirement. The monotonic decrease of the cost-function, however, is a self-evident property of any well-behaved minimization. In operational 4-D assimilation applications we may not have enough computer time to complete the minimization. A monotonic decrease would enable us to terminate minimization at any iteration step, still having confidence that we are closer to the minimum in all aspects (cost-function and gradient norm). It is sometimes suggested that, if the diagonal elements of the Hessian are known, a good preconditioner may be a diagonal matrix, with elements equal to the diagonal elements of the Hessian. In fact, Forsythe and Strauss (1955) have shown that a diagonal matrix with diagonal elements of the Hessian as its elements, is optimal among all diagonal preconditioners. This approach is essentially pursued by Nocedal (1980), Liu and Nocedal (1988) and Nash (1985). Encouraging results, obtained using a similar approach in application to 4-D variational data assimilation, are recently reported by Courtier et al. (1992). It is generally suggested that diagonal preconditioning could be very successful in large-scale minimization problems (e.g., Gill et al., 1981; Luenberger, 1984). In other words, a simple rescaling of variables may significantly improve the minimization.

In 4-D variational data assimilation problems, the cost-function in the form

$$J(\boldsymbol{x})=\tfrac{1}{2}\sum_{n}(M(\boldsymbol{x})-\boldsymbol{a})^{\mathrm{T}}\,\boldsymbol{W}(M(\boldsymbol{x})-\boldsymbol{a}) \qquad (1.1)$$

is usually minimized. Here M represents the

nonlinear forecast model operator, $\boldsymbol{x}$ is control variable (initial conditions in this case), $\boldsymbol{a}$ refers to data, and $\boldsymbol{W}$ is the weight assigned to data. The index "n" refers to observational times. The corresponding Hessian matrix can be calculated (Thacker, 1989; Rabier and Courtier, 1992) from

$$\boldsymbol{H} = \sum_n \boldsymbol{M}'^{\mathrm{T}} \boldsymbol{W} \boldsymbol{M}' + \sum_n (\boldsymbol{M}'')^{\mathrm{T}} \boldsymbol{W}(\boldsymbol{M}(\boldsymbol{x}) - \boldsymbol{a}), \tag{1.2}$$

where $\boldsymbol{M}'$ is the tangent linear (sometimes called the resolvent), and $\boldsymbol{M}'^{\mathrm{T}}$ its adjoint operator. Operator $(\boldsymbol{M}'')^{\mathrm{T}}$ is the adjoint of a second derivative of the forecast model with respect to the control variable. The Hessian is clearly not a diagonal matrix, but it is positive definite (assuming the minimum exists) and symmetric (e.g., Thacker, 1989; Wang et al., 1992). For quadratic cost-function the solution of the minimization problem (1.1) is equivalent to the solution of Newton's equation

$$\boldsymbol{H}\boldsymbol{d} = -\boldsymbol{g}, \tag{1.3}$$

where $\boldsymbol{d}$ is the descent direction (unknown), and $\boldsymbol{g}$ is the (known) gradient of the cost-function J. The Hessian matrix is defined at the solution. Note that the second term on the right-hand side of (1.2) is zero for linear forecast model and could be neglected at points near the solution. Then, the Hessian is

$$\boldsymbol{H} = \sum_n \boldsymbol{M}'^{\mathrm{T}} \boldsymbol{W} \boldsymbol{M}'. \tag{1.4}$$

This is an approximation suggested in the Gauss-Newton minimization method (e.g., Gill et al., 1981; Hoffman, 1986). Also, one may think of employing the Hessian (1.4) in preconditioning. In that case, a truncated quasi-Newton method (e.g., Nash, 1985; Wang and Navon, 1992) may be used. In each iteration of minimization, a linear subproblem is solved, given by Newton's equation at the current iteration

$$\boldsymbol{H}_k \boldsymbol{d}_k = -\boldsymbol{g}_k \tag{1.5}$$

where index "k" refers to iteration number. For example, one may use a few iterations of the linear conjugate gradient algorithm (Gill et al., 1981). The application of (1.4) to a vector involves simply one forward integration of the linear tangent model, followed by a backward adjoint integration. For large scale minimization problems the memory requirements may not be an obstacle if the conjugate gradient method is used, but the computational cost of such designed method is still too high. For example, five iterations of the conjugate gradient algorithm, with the Hessian (1.4) used in (1.5), would be equivalent to 10 integrations of the linear tangent model for the length of assimilation period. However, if the compter time is not a limiting factor, the Hessian in the form (1.4) may be used as a perfect preconditioner in "twin experiment" assimilation, where "data" are generated by the forecast model itself. Unfortunately, experiments we have conducted with operationally obtained OI analyses using this approach in the preconditioning procedure, have confirmed that the second term in (1.2) cannot be neglected. To calculate the Hessian from the complete form of (1.2) is, at present, computationally prohibitive in realistic 4-D variational data assimilation.

In this paper we propose a preconditioning based on a use of a Taylor series expansion and a requirement that the expected decrease of the cost-function is balanced by a preconditioned gradient norm decrease. The preconditioning matrix is positive definite and diagonal. This new preconditioner was tested in two realistic cases of the 4-D variational data assimilation. The results were compared with results obtained using another diagonal preconditioner that worked extremely well in these cases. The new preconditioning has proven superior in all aspects. Overall results show smooth and monotonic decrease of both the cost-function and the gradient norm, in only a few iterations. The proposed preconditioning was used with the memoryless quasi-Newton method of Shanno (1978), and a restart procedure defined in Shanno (1985), but it could be easily employed in other minimization methods. The procedure is inexpensive to apply, and has a potential in practical applications to large scale problems of 4-D variational data assimilation.

In Section 2 we describe the proposed preconditioner. In Section 3 we discuss the conducted experiments, results are presented and discussed in Section 4, and concluding remarks are given in Section 5.

2. The preconditioning method

2.1. Mathematical derivation

In subsequent mathematical derivation we will implicitely assume a grid point model space, and an Euclidean basis. Suppose that the preconditioning matrix $\boldsymbol{P}$ is given by

$$\boldsymbol{P} = \boldsymbol{E}\boldsymbol{E}^{\mathrm{T}} \tag{2.1}$$

where

$$\boldsymbol{E} = \boldsymbol{W}^{1/2}\boldsymbol{G}^{-1/2}. \tag{2.2}$$

Here, $\boldsymbol{W}$ is a weight matrix, and $\boldsymbol{G}$ is a matrix to be determined. Both $\boldsymbol{W}$ and $\boldsymbol{G}$ are assumed to be diagonal, implying that $\boldsymbol{E}$ and $\boldsymbol{P}$ are diagonal. Also, $\boldsymbol{W}$ and $\boldsymbol{G}$ are both self-adjoint, since they are defined over the real field. The reason for choosing this particular form for preconditioning matrix is that the weight matrix $\boldsymbol{W}$ is related to the Hessian (eq. (1.2)), a "perfect" preconditioner. The matrix $\boldsymbol{W}$ is an approximation of the Hessian matrix for $\boldsymbol{M}' = \boldsymbol{I}$. The elements of the matrix $\boldsymbol{G}$ may be viewed as refined scale factors correcting for "rough" scaling introduced by $\boldsymbol{W}$ (e.g., Navon and de Villiers, 1983). Thus, the matrix $\boldsymbol{G}$ represents an influence of the model dynamics on preconditioning. The dispersive and amplifying effects of the forecast model cause an additional stretching of the hyperellipsoid of the cost-function isopleths. It is the matrix $\boldsymbol{G}$ which should be able to account for this effect, making the isopleths more spherical. The matrix $\boldsymbol{W}$ is positive definite, and its elements are presumed known. The elements of the matrix $\boldsymbol{G}$ are denoted γ, and one may write

$$\boldsymbol{G} = \mathrm{diag}(\gamma_L^c) \tag{2.3}$$

Here, L refers to the vertical levels of the numerical model, and the index "c" denotes the component of the control variable. For example, if the control variable is defined as initial conditions for the forecast model, "c" would refer to a prognostic variable, such as: surface pressure, temperature, winds, etc. In the case of the spectral model the index "c" would refer to spectral coefficients. The value of an element of the matrix $\boldsymbol{G}$ is the same for a given control variable component, at all horizontal points at the vertical level L. Note, for example, that there will be only one value of γ for surface pressure because there are no vertical levels involved. One can write

$$\gamma_L^c = (\gamma^{\mathrm{ps}}, \gamma_1^{\mathrm{t}}, \ldots, \gamma_{LM}^{\mathrm{t}}, \gamma_1^{\mathrm{u}}, \ldots, \gamma_{LM}^{\mathrm{u}}, \gamma_1^{\mathrm{v}}, \ldots, \gamma_{LM}^{\mathrm{v}}) \tag{2.4}$$

where LM is the number of vertical levels in the forecast model, and superscripts "ps", "t", "u" and "v" refer to surface pressure, temperature, and wind components, respectively. In the rest of the paper the superscript "c" will be omitted in the notation for the elements of matrix $\boldsymbol{G}$. Unless specified otherwise, it will be assumed that the elements of the matrix $\boldsymbol{G}$ are defined for specific variable at the specific vertical level. The actual design of preconditioning will include calculation of unknown elements of the matrix $\boldsymbol{G}$.

The reason for choosing the particular structure of the matrix $\boldsymbol{G}$ (2.3) and its elements (2.4) is due to the structure observed in the adjustment of the control variable in the first iteration. The changes in the control variable experience large variability by component and by vertical level. By improving the scaling for each variable, and for each vertical level, we expect to improve the preconditioning. For large scale minimization problems, it is easier to estimate a global adjustment (over many points at certain vertical level) than to make a local estimate, at each point. Note that, in general, it may be adequate to chose a different type of variability of the control variable, related to the problem at hand. For example, if there is no significant variability in the vertical, the form (2.3) may be reduced to the number of prognostic variables only.

At any iteration of the minimization the update of the control variable $\boldsymbol{x}$ is given by

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha\, \boldsymbol{d}, \tag{2.5}$$

where index "k" refers to iteration number, $\boldsymbol{d}$ is a descent direction (as before), and α is a step-length. The descent algorithm is usually related to the currently computed gradient, and information about the gradient and descent direction from previous iterations. In the first iteration of most minimization algorithms however, the only information one has is the initial gradient. Then, the initial descent direction is given by a preconditioned steepest descent method, i.e.,

$$\boldsymbol{d} = -\boldsymbol{P}^{-1}\boldsymbol{g}, \tag{2.6}$$

where $\boldsymbol{P}$ is a general preconditioning matrix (not necessarily the same as in (2.1–2.2)).

Near the initial first guess, we expect that the Taylor series expansion for the general cost-function F is valid

$$F_{k+1} = F_k + \alpha \boldsymbol{g}^{\mathrm{T}} \boldsymbol{d} + \tfrac{1}{2}\alpha^2 \boldsymbol{d}^{\mathrm{T}} \boldsymbol{H}\boldsymbol{d}. \tag{2.7}$$

Here, $\boldsymbol{d}^{\mathrm{T}} \boldsymbol{H}\boldsymbol{d}$ and $\boldsymbol{g}^{\mathrm{T}} \boldsymbol{d}$ represent Euclidean inner products defined over the entire model space. Employing Newton's equation (1.3) and eq. (2.7), and introducing $\Delta F = F_k - F_{k+1}$ (a *decrease* of the cost function in the first iteration), we obtain

$$\Delta F = \alpha \left(1 - \frac{\alpha}{2}\right) \boldsymbol{d}^{\mathrm{T}} \boldsymbol{H}\boldsymbol{d}. \tag{2.8}$$

This expression is similar to the scaling of conjugate gradient introduced by Fletcher (1972), except here an accurate line search is not assumed. After introducing

$$\eta = \frac{\boldsymbol{d}^{\mathrm{T}} \boldsymbol{W}\boldsymbol{d}}{\boldsymbol{d}^{\mathrm{T}} \boldsymbol{H}\boldsymbol{d}}, \tag{2.9}$$

the relation (2.8) becomes

$$\Delta F = \alpha \left(1 - \frac{\alpha}{2}\right) \frac{1}{\eta} (\boldsymbol{d}^{\mathrm{T}} \boldsymbol{W}\boldsymbol{d}). \tag{2.10}$$

Let us define a subspace of the model space over all horizontal points, at the particular vertical level and for the particular control variable. Now, let us make an assumption that the Taylor expansion (2.10) is satisfied also at each vertical level and for each control variable, i.e.,

$$\Delta F_L = \alpha \left(1 - \frac{\alpha}{2}\right) \frac{1}{\eta} (\boldsymbol{d}^{\mathrm{T}} \boldsymbol{W}\boldsymbol{d})_L, \tag{2.11}$$

where L refers to the vertical level. This is an assumption which will be tested by the results of minimization with preconditioning. The inner product $(\boldsymbol{d}^{\mathrm{T}} \boldsymbol{W}\boldsymbol{d})_L$ is now defined over all points representing a chosen subspace. Similarly, ΔF_L represents a decrease of the cost-function over all points of the subspace. Some characteristics of the parameter η are discussed in the Appendix. It appears that the value of η is not changing significantly from case to case. The assumption (2.11) is critical, because it allows the cost-function decrease to be expressed in terms of a diagonal matrix, $\boldsymbol{W}$, instead of a general symmetric matrix, $\boldsymbol{H}$. Several other diagonal matrices were tried instead of $\boldsymbol{W}$, but with no success. It appears that the choice should be based on the definition on the Hessian matrix (1.2) as much as possible.

Starting with (2.6), one can consider three important cases:

$$\boldsymbol{d} = -\boldsymbol{H}_{\mathrm{D}}^{-1} \boldsymbol{g}, \tag{2.12}$$

$$\boldsymbol{d} = -\boldsymbol{W}^{-1} \boldsymbol{g}, \tag{2.13}$$

$$\boldsymbol{d} = -\boldsymbol{G}\boldsymbol{W}^{-1} \boldsymbol{g}, \tag{2.14}$$

where $\boldsymbol{H}_{\mathrm{D}}$ is an optimal diagonal preconditioner, with elements equal to diagonal elements of the Hessian. Thus, the first choice, (2.12), is the best descent direction one can get with diagonal preconditioning matrix. The second choice, (2.13), is a "natural" initial choice because there is a need for scaling of variables with different physical units, and also because it is indirectly related to the Hessian matrix (1.2). The choice (2.14) introduces another diagonal matrix $\boldsymbol{G}$, and corresponds to the choice (2.1–2.3) for the preconditioning matrix. Note that all considered matrices are diagonal. Also, recall that all diagonal matrices are commutative. Employing all three possibilities for the descent direction in (2.11), and using properties of real-valued diagonal matrices

$$\Delta F_L = \alpha \left(1 - \frac{\alpha}{2}\right) \frac{1}{\eta} (\boldsymbol{g}^{\mathrm{T}} \boldsymbol{H}_{\mathrm{D}}^{-1} \boldsymbol{W} \boldsymbol{H}_{\mathrm{D}}^{-1} \boldsymbol{g})_L, \tag{2.15}$$

$$\Delta F_L = \alpha \left(1 - \frac{\alpha}{2}\right) \frac{1}{\eta} (\boldsymbol{g}^{\mathrm{T}} \boldsymbol{W}^{-1} \boldsymbol{g})_L, \tag{2.16}$$

$$\Delta F_L = \alpha \left(1 - \frac{\alpha}{2}\right) \frac{1}{\eta} (\boldsymbol{g}^{\mathrm{T}} \boldsymbol{G}\boldsymbol{W}^{-1} \boldsymbol{G}\boldsymbol{g})_L. \tag{2.17}$$

Note that the corresponding cost-function decrease (left-hand side of eqs. (2.15–17)) may differ very much, depending on the chosen preconditioning. The idea is to choose the elements of the matrix $\boldsymbol{G}$ in such way that the optimal decrease of the cost-function (2.15) is achieved also in (2.17).

Let $\{\boldsymbol{e}_i\}$ denote a unique Euclidean basis of all diagonalizable linear operators over the

N-dimensional real field (e.g., Hoffman and Kunze, 1971). Also, let

$$\boldsymbol{H}_{\mathrm{D}} = \mathrm{diag}(h_i), \qquad \boldsymbol{W} = \mathrm{diag}(w_i), \qquad \boldsymbol{G} = \mathrm{diag}(\gamma_i). \tag{2.18}$$

Using the fact that the eigenvalues of any diagonal matrix are precisely those diagonal elements, it follows from (2.15) and (2.18) that

$$\Delta F_L = \alpha\left(1-\frac{\alpha}{2}\right)\frac{1}{\eta}\sum_{i(L)}(w_i h_i^{-1})^2(w_i^{-1}(\boldsymbol{g}\boldsymbol{e}_i^{\mathrm{T}})^2). \tag{2.19}$$

Here, index "$i(L)$" denotes all horizontal points at the level L, for a particular model variable. Similarly, from (2.17) and (2.18)

$$\Delta F_L = \alpha\left(1-\frac{\alpha}{2}\right)\frac{1}{\eta}\gamma_L^2\sum_{i(L)} w_i^{-1}(\boldsymbol{g}\boldsymbol{e}_i^{\mathrm{T}})^2, \tag{2.20}$$

where a constant value for the γ was assumed at the level L. The requirement that the cost-function decrease in (2.20) is equal to the optimal decrease (2.19) results in

$$\gamma_L = \{\overline{(w_i h_i^{-1})^2}\}^{1/2}, \tag{2.21}$$

where the overbar denotes a weighted average, defined as

$$\overline{(\ldots)} = \frac{\sum_{i(L)}(\ldots)\, w_i^{-1}(\boldsymbol{g}\boldsymbol{e}_i^{\mathrm{T}})^2}{\sum_{i(L)} w_i^{-1}(\boldsymbol{g}\boldsymbol{e}_i^{\mathrm{T}})^2}. \tag{2.22}$$

Unfortunately, eigenvalues of $\boldsymbol{H}_{\mathrm{D}}$ are usually not known, and (2.21) cannot be used to determine unknown elements of the matrix $\boldsymbol{G}$. In order to obtain γ let us proceed from (2.20), assuming that a cost-function decrease on the left-hand side is optimal. Then, solving for γ

$$\gamma_L^2 = \frac{\eta}{\alpha(1-\alpha/2)}\frac{\Delta F_L}{(\boldsymbol{g}^{\mathrm{T}}\boldsymbol{W}^{-1}\boldsymbol{g})_L}, \tag{2.23}$$

where

$$(\boldsymbol{g}^{\mathrm{T}}\boldsymbol{W}^{-1}\boldsymbol{g})_L = \sum_{i(L)} w_i^{-1}(\boldsymbol{g}\boldsymbol{e}_i^{\mathrm{T}})^2. \tag{2.24}$$

Let us introduce the relative decrease of the cost-function, k,

$$k = \frac{\Delta F}{F}, \tag{2.25}$$

where F refers to the total cost-function (over the entire model space). In order to keep the problem tractable, let us assume

$$\Delta F_L = kF_L, \qquad \text{for all } L. \tag{2.26}$$

The assumption (2.26) implies that the expected *relative* decrease of the cost-function is same at each level, and for each variable. The assumption is introduced because it seems easier to make an accurate estimate of a single parameter k, a relative decrease of the total cost-function, rather than to estimate a relative decrease at each vertical level. Employing (2.26), eq. (2.23) becomes

$$\gamma_L^2 = \frac{\eta k}{\alpha(1-\alpha/2)}\frac{F_L}{(\boldsymbol{g}^{\mathrm{T}}\boldsymbol{W}^{-1}\boldsymbol{g})_L}. \tag{2.27}$$

Given the optimal decrease of the cost-function over the entire model space (2.10), one can define a new, constant parameter γ_0, such that (2.10) becomes

$$\Delta F = \alpha\left(1-\frac{\alpha}{2}\right)\frac{1}{\eta}\gamma_0^2(\boldsymbol{g}^{\mathrm{T}}\boldsymbol{W}^{-1}\boldsymbol{g}). \tag{2.28}$$

From (2.25) and (2.28) one obtains

$$\frac{\eta k}{\alpha(1-\alpha/2)} = \gamma_0^2\frac{(\boldsymbol{g}^{\mathrm{T}}\boldsymbol{W}^{-1}\boldsymbol{g})}{F}. \tag{2.29}$$

From (2.27) and (2.29), it follows:

$$\gamma_L^{\mathrm{c}} = \left\{\gamma_0\frac{(\boldsymbol{g}^{\mathrm{T}}\boldsymbol{W}^{-1}\boldsymbol{g})^{1/2}}{F^{1/2}}\right\}\frac{[F_L^{\mathrm{c}}]^{1/2}}{[(\boldsymbol{g}^{\mathrm{T}}\boldsymbol{W}^{-1}\boldsymbol{g})_L^{\mathrm{c}}]^{1/2}}. \tag{2.30}$$

Eq. (2.30) is an expression for the elements of a diagonal matrix $\boldsymbol{G}$, needed to be calculated for each component of the control variable (denoted by index "c"), at each vertical level. The index "c" has been used again in order to underline the significance of calculating the elements of $\boldsymbol{G}$ for each variable. The first term in brackets, on the right-hand side of (2.30), is a scaling parameter, dependent on a particular case. The last term on the right-hand side of (2.30) represents the (vertical) structure introduced by the preconditioner. If the

line search algorithm does not depend on the initial guess for the step length, both scaling parameters are not important, and only the last term on the right-hand side of (2.30) may be used. However, if the initial guess for the step length is used, the scaling factors could be very important. This is related to a safeguard procedure (Gill et al., 1981), in which a prespecified maximum step-length is imposed. This aspect is discussed in more details later, in Section 4. The parameter γ_0 can be empirically calculated. Unfortunately, the empirical choice of γ_0 will not garantee the optimal reduction of the cost-function in (2.28). On the other hand, for an only slightly incorrect choice of γ_0, the line search procedure would create step-length α such that the decrease of the cost-function is closer to the optimal one, thus making (2.28) closer to the truth. In our experiments (including a number of additional cases not shown here), this seems to be confirmed. A practical advice is to try to choose the value of γ_0 which would allow the optimal step-length to be close to the unit guess value, or slightly larger than one. In operational applications, with a forecast model, weights and a distance of the first guess from the solution not changing significantly from day to day, one can calculate the parameter γ_0 from the previous assimilation cycle. Note that (2.30) is easy to compute, and there is no significant requirements regarding the computer time and memory. The complete preconditioning matrix may be written as

$$\boldsymbol{P}^{-1} = \mathrm{diag}(\gamma_L w_i^{-1}), \tag{2.31}$$

where index "i" refers to a model space point, and γ_L is a corresponding element of the matrix $\boldsymbol{G}$, defined by (2.30). The preconditioning matrix $\boldsymbol{P}$ is positive definite, because both matrices, $\boldsymbol{G}$ and $\boldsymbol{W}^{-1}$ are diagonal and positive definite.

Note that, in the least square problems (4-D variational data assimilation), F is calculated as a sum of squared distances between the model and data, over all times when data were available. Since the cost-function F used in Taylor expansion (2.8) is not directly related to the number of time levels with data, it may be useful to introduce in (2.30) some type of a weighted average over the assimilation period. For example, one may choose

$$F = \frac{\sum_n \mu_n F_n}{\sum_n \mu_n}, \tag{2.32}$$

where F_n is the calculated squared distance between model and data, μ_n is the weight given to F_n, and the index "n" refers to time levels with data. Formula for the elements of the matrix $\boldsymbol{G}$ (2.30) will not be altered by this choice of the cost-function F.

2.2. *Convergence analysis*

Recall that the preconditioned Hessian matrix, $\boldsymbol{H}_\mathrm{p}$, may be written as $\boldsymbol{H}_\mathrm{p} = \boldsymbol{E}^{-1}\boldsymbol{H}\boldsymbol{E}^{-\mathrm{T}}$. When the preconditioning matrix is a diagonal matrix with elements equal to diagonal elements of the Hessian (2.12), the preconditioned Hessian is

$$\boldsymbol{H}_{\mathrm{p}_0} = \boldsymbol{H}_\mathrm{D}^{-1/2}\boldsymbol{H}\boldsymbol{H}_\mathrm{D}^{-\mathrm{T}/2}. \tag{2.33}$$

When there is only a "rough" scaling as preconditioning (2.13), the preconditioned Hessian is

$$\boldsymbol{H}_{\mathrm{p}_1} = \boldsymbol{W}^{-1/2}\boldsymbol{H}\boldsymbol{W}^{-\mathrm{T}/2} \tag{2.34}$$

Finally, when the preconditioning is given by (2.1–2.2), the preconditioned Hessian is

$$\boldsymbol{H}_{\mathrm{p}_2} = (\boldsymbol{G}^{1/2}\boldsymbol{W}^{-1/2})\,\boldsymbol{H}(\boldsymbol{G}^{1/2}\boldsymbol{W}^{-1/2})^{\mathrm{T}}. \tag{2.35}$$

In order to compare the proposed preconditioning with the optimal *diagonal* preconditioning matrix, $\boldsymbol{H}_\mathrm{D}$, let substitute $\boldsymbol{H}$ by $\boldsymbol{H}_\mathrm{D}$ in (2.33–2.35). By doing this, only the effect of preconditioning on diagonal elements of the Hessian is considered, neglecting the off-diagonal elements. Since all matrices, $\boldsymbol{H}_\mathrm{D}$, $\boldsymbol{W}$, and $\boldsymbol{G}$, are diagonal and real-valued, one obtains

$$\boldsymbol{H}_{\mathrm{p}_0} = \boldsymbol{H}_\mathrm{D}^{-1}\boldsymbol{H}_\mathrm{D} = \boldsymbol{I}, \tag{2.36}$$

$$\boldsymbol{H}_{\mathrm{p}_1} = \boldsymbol{W}^{-1}\boldsymbol{H}_\mathrm{D}, \tag{2.37}$$

$$\boldsymbol{H}_{\mathrm{p}_2} = \boldsymbol{G}\boldsymbol{W}^{-1}\boldsymbol{H}_\mathrm{D}. \tag{2.38}$$

Obviously, all elements of the matrix $\boldsymbol{H}_{\mathrm{p}_0}$ are equal to one. For matrices $\boldsymbol{H}_{\mathrm{p}_1}$ and $\boldsymbol{H}_{\mathrm{p}_2}$ one can obtain diagonal elements (eigenvalues) from (2.18)

$$\begin{aligned} \boldsymbol{H}_{\mathrm{p}_1} &= \mathrm{diag}(w_{i(L)}^{-1} h_{i(L)}), \\ \boldsymbol{H}_{\mathrm{p}_2} &= \mathrm{diag}(\gamma_L w_{i(L)}^{-1} h_{i(L)}), \end{aligned} \tag{2.39}$$

where index "$i(L)$" denotes a point at the vertical level L. Properties of a proposed preconditioning matrix will be best seen by comparing the condition number of matrices $\boldsymbol{H}_{\mathrm{p}_1}$ and $\boldsymbol{H}_{\mathrm{p}_2}$, knowing that the optimal condition number is 1. For

notational convenience let consider the condition number of inverse matrices of $\boldsymbol{H}_{p_1}$ and $\boldsymbol{H}_{p_2}$. Recall that the condition numbers of an invertible matrix and its inverse are equal (e.g., Gill et al., 1981). Using the notation

$$\beta_i = w_i h_i^{-1}, \tag{2.40}$$

where index "i" denotes a point in a model space, one obtains for the relevant condition numbers:

$$K(\boldsymbol{H}_{p_0}) = 1, \tag{2.41}$$

$$K(\boldsymbol{H}_{p_1}) = \frac{\beta_{\max}}{\beta_{\min}}, \tag{2.42}$$

$$K(\boldsymbol{H}_{p_2}) = \frac{[\beta/\gamma_L]_{\max}}{[\beta/\gamma_L]_{\min}}. \tag{2.43}$$

Note that eq. (2.21) can be written using (2.40) as

$$\gamma_L = \{\overline{\beta_i^2}\}^{1/2}, \tag{2.44}$$

where the overbar denotes an average over all horizontal points at the vertical level L, and for specific model variable (eq. (2.22)). This indicates that the eigenvalues of the matrix $\boldsymbol{H}_{p_2}$ are obtained by dividing eigenvalues of $\boldsymbol{H}_{p_1}$ by an estimate of their respective average value at given vertical level and for given model variable. This is very important for understanding the effect of proposed preconditioning on the spectrum of the preconditioned Hessian matrix. In large scale minimization problems, even an approximate scaling of eigenvalues of the Hessian may have a significant effect on the speed of convergence. Now, let "l_{max}" be the vertical level where the maximum eigenvalue of $\boldsymbol{H}_{p_2}$ is found, and let "l_{min}" denote the vertical level of the minimum eigenvalue of $\boldsymbol{H}_{p_2}$, for given model variable. Important to note is that these vertical levels do not, in general, correspond to the levels where the extreme eigenvalues of $\boldsymbol{H}_{p_1}$ are located. Using

$$\frac{\beta_{l_{\max}}}{\beta_{l_{\min}}} \leqslant \frac{\beta_{\max}}{\beta_{\min}} = K(\boldsymbol{H}_{p_1}), \tag{2.45}$$

it follows from (2.43)

$$K(\boldsymbol{H}_{p_2}) = \frac{[\beta/\gamma_L]_{\max}}{[\beta/\gamma_L]_{\min}} = \frac{[\gamma_L]_{l_{\min}}}{[\gamma_L]_{l_{\max}}} \frac{\beta_{l_{\max}}}{\beta_{l_{\min}}} \leqslant \frac{[\gamma_L]_{l_{\min}}}{[\gamma_L]_{l_{\max}}} K(\boldsymbol{H}_{p_1}). \tag{2.46}$$

Hence, the condition of the matrix $\boldsymbol{H}_{p_2}$ critically depends on the ratio of γ_L at the vertical levels l_{max} and l_{min}, which indirectly reflects the existing eigenvalue structure of the Hessian. From (2.46), the condition for improved convergence is given by

$$\frac{[\gamma_L]_{l_{\min}}}{[\gamma_L]_{l_{\max}}} < 1. \tag{2.47}$$

Since the levels l_{min} and l_{max} are generally not known, it is difficult to know in advance if the condition (2.47) is satisfied. For practical purposes, one can compute the values of γ_L and see if there is a significant variability by model variable and/or by vertical level. If there is a relative difference of, at least, one order of magnitude, it may be an indication that the proposed preconditioner could perform well. In the experiments discussed in this paper the minimum value of the ratio (2.47) was about 4×10^{-2}. On the other hand, since the preconditioning is quite simple to implement, it may be best to simply try the proposed method in actual minimization.

3. Experimental design

3.1. Models and data

A simplified version of the NMC's new regional forecast model being developed for operational use has been employed. Both the non-linear and the adjoint models have dynamics and horizontal diffusion, but moisture and parameterized processes are not included.

The non-linear forecast model is defined in the eta vertical coordinate (Mesinger et al., 1988; Janjic, 1990). The model employs a split-explicit time differencing, and the time step used in the experiments presented here is 200 s. The horizontal resolution is approximately 80 km, and the domain of integration covers the North-American continent. Two versions of the model have been used, one with 16, and the other with 17 vertical layers. In both versions of the model the top was at 100 hPa. The additional layer helps in better resolution near the bottom of the model. However, the effect of the additional layer will be important mostly in surface processes, not present in the

simplified version of the forecast model used here. Boundary conditions are not adjusted in the adjoint calculations; they are taken from available analyses.

Data used are operationally obtained OI analyses (DiMego, 1988), available every 3 h, in the experiment with 16 vertical levels, and every 6 h in the experiment with 17 vertical levels. Since only a "dry" version of the forecast model (and its adjoint) is employed, data used from the analysis are surface pressure, temperature, and wind.

3.2. *Experiments*

The assimilation period in all experiments was 12 h. Two sets of the 4-D variational data assimilation experiments are performed:

(1) using a 16-level version of the model, on 27 December 1991, from 12–24 UTC;
(2) using a 17-level version of the model, on 30 June 1992, from 00–12 UTC.

A control run in each experiment was designed using $\boldsymbol{G} = \boldsymbol{I}$ (eqs. (2.1–2.2)), so that only a "rough" scaling is kept, based on the weights given to data. Also, a "preconditioned" assimilation was performed for each case, using γ computed from eq. (2.30). Note that the control experiment is also preconditioned, because a general scaling of variables with different physical units is included, in the form of the weight matrix $\boldsymbol{W}$. In both experiments a significant variability of γ was found. For surface pressure the value was about 0.3–0.4; for temperature, γ was in the range from 0.1 at low levels to 0.01 at upper levels; and for winds, the values were from 4×10^{-2} at lower levels to 0.2 at upper levels. These values of γ indicate a significant change of the initially chosen weights. The number of degrees of freedom in this problem is of the order of 4×10^5. Hence, the Hessian matrix has a dimension of $(4 \times 10^5) \times (4 \times 10^5)$. The dimension of the subspace in which the gradient norm and cost-function are calculated in (2.30) is 8581. This is a number of horizontal points at each vertical level and for each variable. All these numbers suggest that the problem of preconditioning in realistic data assimilation experiments is quite challenging.

The cost-function was defined as in Z93, with a penalty term controlling the amount of gravity waves

$$J = \frac{1}{2} \sum_t (\boldsymbol{x}^t - \boldsymbol{x}^t_{\text{obs}})^{\mathrm{T}} \boldsymbol{W} (\boldsymbol{x}^t - \boldsymbol{x}^t_{\text{obs}}) + \frac{1}{2} \sum_n \varepsilon \left(\frac{\partial \mathbf{Div}}{\partial t} \right)^{\mathrm{T}}_n \left(\frac{\partial \mathbf{Div}}{\partial t} \right)_n \tag{3.1}$$

where index "t" refers to observational times, and index "n" to time steps of the forecast model. The analysis weight matrix is denoted $\boldsymbol{W}$, the relative weight given to the penalty term ε, and horizontal divergence is denoted **Div**. Since data used are OI analyses, all weights μ in (2.32) are assumed equal, and F is simply a mean value of the calculated cost function over the assimilation period. In all experiments the weight matrix $\boldsymbol{W}$ is assumed diagonal, thus the preconditioning matrix $\boldsymbol{P}$ (eq. (2.1)) was also a diagonal matrix. The choice of weights was based on a typical radiosonde observational error variance, same as used at NMC operationally. In order to account for additional analysis error, the variance was multiplied by a factor of two. Although this is only a rough approximation of the true analysis error, the calculated weights show a vertical structure typical for real data. Weights for temperature vary from $0.083\ \mathrm{K}^{-2}$ at the bottom to $0.23\ \mathrm{K}^{-2}$ at the top. For winds, weights are $0.049\ \mathrm{m}^{-2}\,\mathrm{s}^2$ at the bottom, and $0.31\ \mathrm{m}^{-2}\,\mathrm{s}^2$ at the top of the model. The weight given to surface pressure is $0.0002\ \mathrm{Pa}^{-2}$.

The minimization algorithm employed is a memoryless quasi-Newton algorithm of Shanno (1978), with a restart procedure providing global convergence (Shanno, 1985). Our experience is that this algorithm is well suited for 4-D variational data assimilation (also, see Navon and Legler, 1987).

In the experiments presented in this paper, we have used $\gamma_0 = 0.0227$.

4. Results

All results are assessed by comparing the behaviour of the cost-function and the gradient norm during minimization.

4.1. *The cost-function*

In Figs. 1 and 2, the decrease of the normalized total cost-function during minimization is shown: in Fig. 1 the results of experiment (1), and in Fig. 2 the results of experiment (2). The dashed line

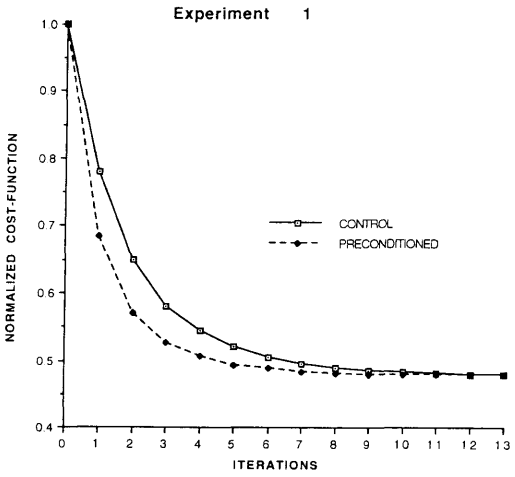

Fig. 2. Same as in Fig. 1, except for experiment (2), with 17-layer model version employed.

represents minimization with the new preconditioner, and the full line the control experiment.

First, note from both figures that the decrease in the control experiment is extremely good. The saturation is reached after only 4–5 iterations of the minimization algorithm. Even so, the decrease of the cost-function in the experiments with new preconditioning is consistently better, in both cases. Most of the improvement occurs in the first 2–3 iterations. From the perspective of the new preconditioner, which includes a refined scaling at vertical levels, it is interesting to look at the decrease of the components of the cost-function as a function of vertical levels. Just to illustrate the effect of a new preconditioner at different vertical levels, we show the relative decrease of the temperature and u-wind cost-function components in the first iteration, for experiment (1). In Fig. 3 the percentage decrease of the temperature component of the cost-function is plotted at different (eta) vertical levels, and in Fig. 4 the corresponding u-wind component is shown. As in the previous figures, the results of the preconditioned minimization are denoted by a dashed line, and the results of the control experiment are denoted by a full line. Positive values represent a decrease of the cost-function, and negative an increase. The temperature cost-function shows an increase at levels 4 and 5, in the control experiment (Fig. 3). With pre-

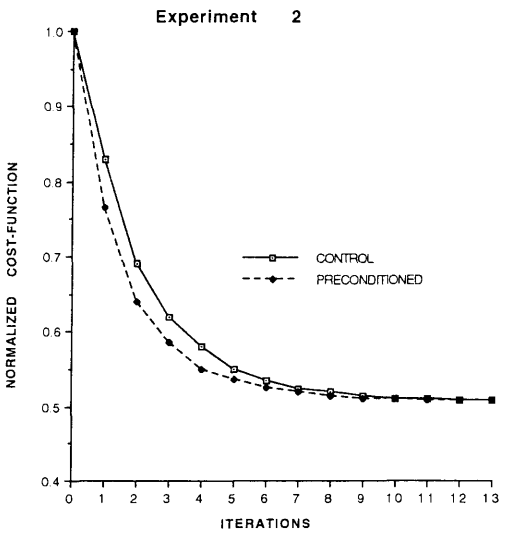

Fig. 1. Normalized cost-function decrease minimization in the experiment (1), with a 16-layer model version employed. The total cost-function is normalized by its initial value. The full line represents the control experiment results, with "rough" scaling only, and the dashed line represents the results of the experiment with the new preconditioning.

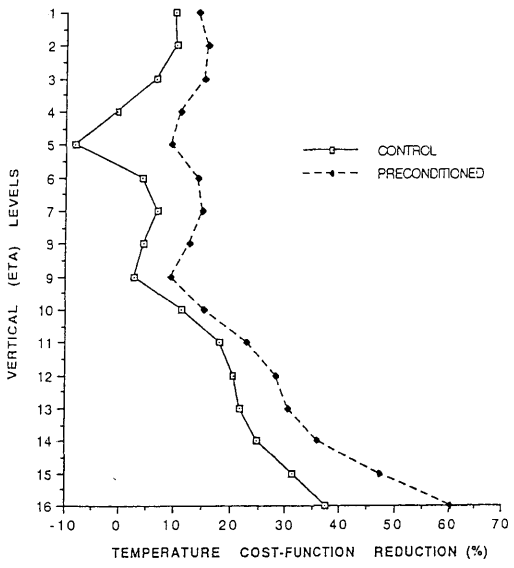

Fig. 3. The % reduction of the temperature part of the cost-function in the first iteration as a function of vertical eta levels, for experiment (1). Note that level 1 corresponds to a top of the model (100 hPa), and level 16 refers to the bottom of the model. Positive values represent a reduction, and negative values represent an increase of the cost-function in first iteration.

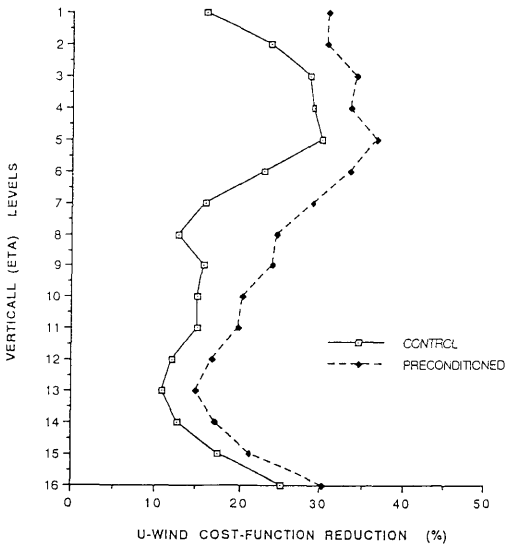

Fig. 4. Same as in Fig. 3, except for the u-wind part of the cost-function.

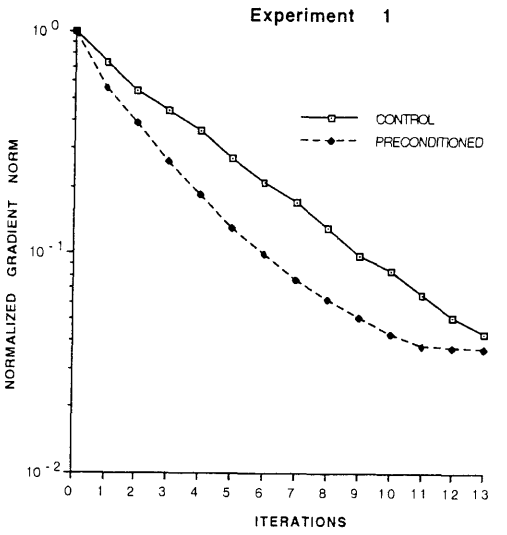

Fig. 5. The normalized gradient norm as a function of iterations in experiment (1), with 16-layer model employed. The full line refers to the control experiment results, and the dashed line to the experiment with the new preconditioning.

conditioning, a decrease of the cost-function was found at all levels. Both figures indicate quite significant improvement. It appears that the new preconditioner has the effect of making a more balanced adjustment at each vertical level.

4.2. Gradient norm

Gradient norms normalized by their initial value as functions of the iteration number in experiments (1) and (2) are shown in Figs. 5 and 6, respectively. These figures indicate the convergence rate of minimization, measured by the gradient norm (another way is to measure the cost-function decrease). The results of the control experiments are denoted by a full line, and a dashed line represents results with the new preconditioning. Good convergence properties (Figs. 5 and 6) confirm superior results of the proposed preconditioning. Saturation is reached after only 11–13 iterations. There is a similar total decrease of the gradient norm, of about two orders of magnitude. Note smooth behaviour of the gradient norm during minimization, in both the control and preconditioning experiments, very important in practical applications of minimization.

An important point to be stressed is related to a safeguard procedure in the minimization, involving a definition of an upper-limit for a maximum step-length allowed (e.g., Gill et al., 1981). In practical applications, there is a need for a maximum step-length to be defined, so that the minimization update will allow only values that do not undermine significantly the forward model integration.

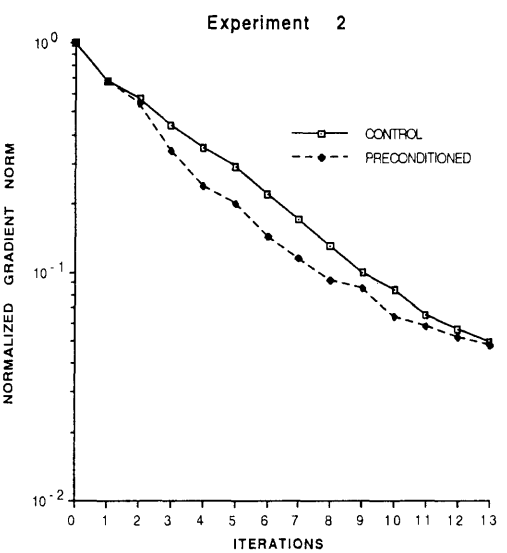

Fig. 6. Same as in Fig. 5, except for experiment (2), with 17-layer model employed.

For example, in 4-D variational data assimilation, one would not allow a 100°C temperature change at a grid point, because it is not physically meaningful, and it could create an extremely large amount of gravity waves in the forward integration. The problem that arises in "rough" scaling (the control experiment) is that the maximum change at one point usually reaches that limit, and the step-length allowed has to be reduced. In a few experiments with different maximum step-lengths (not shown here) it was noted that the minimization could be significantly worsened. Even an increase of the cost-function was found at certain vertical levels. The benefit of the new preconditioner is in reducing the amount of actual change, so that the problem of maximum step length may rarely be encountered. By preconditioning, we change the direction of descent, so that a larger reduction of the cost-function is obtained with smaller adjustment. The preconditioned gradient norm (not shown) is always smaller than in the control experiment. Overall, the preconditioned gradient norm is 4–7 times smaller than in the control experiment, implying that the optimal step-length can be larger in preconditioning experiment. It is the scaling introduced in (2.30) which allows the initial step-length to be determined freely, without using the prescribed upper-limit.

4.3. *Estimate of the condition number of the Hessian matrix*

An attempt was made to calculate the condition number of the Hessian matrix by applying the power and shifted power iteration method (e.g., Wang et al., 1993; Golub and Van Loan, 1989). Using this method it is possible to obtain extremal eigenvalues of a matrix, and then use their ratio to estimate the condition number. The power method involves a Hessian-vector product, computationally demanding task for a large-dimensional problem considered here. A finite-difference approach (e.g., Wang et al., 1993) was used to calculate the matrix-vector product. Due to extremely large computer time requirement, only calculation of the extremal eigenvalues at the beginning and at the end of minimization was attempted. Unfortunately, very slow convergence was found when applying the shifted power method, for calculation of minimum eigenvalues. An estimate of the error of minimum eigenvalue (e.g., Golub and Van Loan, 1989) appeared to be of the same order of magnitude as the eigenvalue itself. The problem was especially difficult in the case of a 17-layer model. The estimate of the maximum eigenvalue of the Hessian was relatively good in both experiments, with an error of the order of 1–2%. Just for illustrative purposes, the results for the 16-layer model will be discussed. At the beginning of minimization, in the control experiment, the maximum eigenvalue estimate was 203, while with preconditioning the eigenvalue was reduced to only 5.9. For the minimum eigenvalue only a rough estimate was possible. In the control experiment the minimum eigenvalue estimate was 1.26, while with preconditioning the value 0.083 was obtained. The resulting condition numbers of the Hessian matrix at the beginning of minimization were 159 in control, and 71 in preconditioning experiment. These values of the condition number, although only roughly estimated, will be used to illustrate the impact of proposed preconditioning on the rate of convergence of the conjugate-gradient (C-G) method. From an estimate of the rate of convergence of C-G method, one can derive the relation between the number of iterations, p, and given convergence criterion ε (e.g., Axelsson and Barker, 1984)

$$p(\varepsilon) \leqslant \frac{1}{2}\sqrt{K(\boldsymbol{H})}\ln\left(\frac{2}{\varepsilon}\right) + 1. \tag{4.1}$$

The convergence criteria is defined as a ratio between the cost-function at the solution, and at the initial point of minimization. In Table 1 the number of iterations of the C-G method is shown as a function of convergence criteria, calculated using (4.1). A potential benefit of preconditioning in C-G method is easily seen.

Table 1. *An estimate of the number of iterations of a conjugate-gradient method for given convergence criteria* ε

ε	No. iterations	
	control experiment	preconditioning experiment
0.5	10	7
10^{-1}	19	13
10^{-2}	34	23
10^{-3}	48	33
10^{-4}	63	42
10^{-5}	77	52

At the end of minimization, a reduction of the condition number of the Hessian is found in both experiments, in agreement with results obtained by Wang et al. (1993). The value of the maximum eigenvalue decreased to 6.2 in control, and to 5.6 in preconditioning experiment. Especially large is the decrease in the control experiment, by a factor of 30. Although much smaller, the decrease in preconditioning experiment was also noticeable. One can interpret this results as a contraction of the cost-function isopleths in the direction of largest eigenvalues of the Hessian. At the same time the minimum eigenvalues were difficult to estimate, due to large errors. As suggested by Wang et al. (1993) one should not expect large adjustments of smallest eigenvalues in minimization, thus the effect of preconditioning on the cost-function isopleths in the direction of smallest eigenvalues may not be significant.

5. Concluding remarks

A new preconditioning is proposed, aimed for use in conjugate-gradient and memoryless quasi-Newton algorithms. However, it can be easily used in any minimization algorithm if the preconditioned steepest descent is used in the first iteration. The new preconditioner is expected to be most beneficial in large scale minimization problems, such as the 4-D variational data assimilation in meteorology and oceanography. The proposed preconditioning is based on the requirement that the control variable adjustment is balanced by the calculated gradient norm. In addition, a second order Taylor series expansion is assumed valid in the vicinity of the first guess (analysis at initial time). The preconditioning matrix is strictly positive definite and diagonal.

The new preconditioning was tested in two cases of the 4-D variational data assimilation experiments with the NMC's new regional forecast model, and with operationally obtained OI analyses. The forecast model and the adjoint have been used in a simplified form, having included dynamics and horizontal diffusion, but no moisture and parameterized processes. A significant improvement was found in all experiments. The vertical structure of the cost-function reduction shows an improvement at all levels. A decrease of the cost-function and the gradient norm is smooth and well-behaved during the minimization, encouraging from the perspective of an operational use. However, one should note that all results are obtained for a specific forecast model and data base. Further testing is needed with other forecast models.

The proposed procedure is very easy to compute and implement, and the computer time and memory requirements are negligible compared to gradient calculation. The method is quite general, and it does not depend on the particular form of the cost-function. It requires, however, that both the cost-function (data) and the gradient be defined in the same space. Therefore, additional assumptions may be needed when applying the proposed preconditioning to direct observations. An inclusion of a quadratic background term in the cost-function (e.g., Lorenc, 1986), however, should not increase the complexity of the preconditioning problem, because the total Hessian has a well-known structure, and the methods to satisfactory solve this problem exist (e.g., Luenberger, 1984). In this paper, all experiments were conducted by adjusting the initial conditions only. Note that there should be no special difficulty in implementing the method to other choices of control variable, such as a model bias correction parameter (Derber, 1989; Z93), boundary conditions, etc.

6. Acknowledgments

I would like to thank John Derber and Dave Parrish for their advices and thoughtful reviews. Also, I would like to thank Jim Purser and two unknown reviewers for their critical and very helpful reviews. My gratitude is extended to University Corporation for Atmospheric Research (UCAR) Visitors Program, for its help and financial support.

Appendix

Mathematical behaviour of the parameter η

By eq. (2.9) from the text, the parameter η is introduced

$$\eta = \frac{\boldsymbol{d}^{\mathrm{T}} \boldsymbol{W} \boldsymbol{d}}{\boldsymbol{d}^{\mathrm{T}} \boldsymbol{H} \boldsymbol{d}}. \tag{A.1}$$

The inner products in (A.1) are defined over all points. Since both the weight matrix $\boldsymbol{W}$ and the Hessian matrix $\boldsymbol{H}$ are positive definite and symmetric, their eigenvalues are real and positive. Also, the corresponding eigenvectors of each matrix form an orthonormal basis in the space χ of the control variable (for example, Hoffman and Kunze, 1971). Let λ denote the eigenvalues of $\boldsymbol{H}$, and w the eigenvalues of $\boldsymbol{W}$. Let $\{\varphi\}$ be the eigenvectors of the Hessian matrix $\boldsymbol{H}$, and $\{\boldsymbol{e}\}$ the eigenvectors of the weight matrix $\boldsymbol{W}$, both defined over the same space χ. Then, for any vector $\boldsymbol{x}$ in χ (see also Thacker, 1989)

$$\boldsymbol{H}\boldsymbol{x} = \sum_{i=1}^{N} \lambda_i (\boldsymbol{x}\varphi_i^{\mathrm{T}})\, \varphi_i, \tag{A.2}$$

where N is the dimension of the space χ. Similarly, for the weight matrix $\boldsymbol{W}$

$$\boldsymbol{W}\boldsymbol{x} = \sum_{i=1}^{N} w_i (\boldsymbol{x}\boldsymbol{e}_i^{\mathrm{T}})\, \boldsymbol{e}_i. \tag{A.3}$$

Using (A.1), (A.2), and the orthogonality between the eigenvector components, it follows from (A.1):

$$\eta = \frac{\sum_i w_i (\boldsymbol{d}\boldsymbol{e}_i^{\mathrm{T}})^2}{\sum_i \lambda_i (\boldsymbol{d}\varphi_i^{\mathrm{T}})^2}. \tag{A.4}$$

The adjustment of the control variable in the first iteration is represented by a vector $\boldsymbol{d}$. Since the basis vectors, $\{\boldsymbol{e}\}$ and $\{\varphi\}$, are orthonormal vectors, one can rewrite (A.4) as

$$\eta = \frac{\sum_i w_i ([\boldsymbol{d}^{\mathrm{T}}\boldsymbol{d}] + 1 - 2[\boldsymbol{d}^{\mathrm{T}}\boldsymbol{d}]^{1/2} \cos(b_i))}{\sum_i \lambda_i ([\boldsymbol{d}^{\mathrm{T}}\boldsymbol{d}] + 1 - 2[\boldsymbol{d}^{\mathrm{T}}\boldsymbol{d}]^{1/2} \cos(a_i))}, \tag{A.5}$$

where "a_i" is an angle between $\boldsymbol{d}$ and φ_i. Similarly, "b_i" is an angle between $\boldsymbol{d}$ and $\boldsymbol{e}_i$. The parameter η may be viewed as a function of $\boldsymbol{d}^{\mathrm{T}}\boldsymbol{d}$. Note that the norm of $\boldsymbol{d}$ is $\|\boldsymbol{d}\| = [\boldsymbol{d}^{\mathrm{T}}\boldsymbol{d}]^{1/2}$. An examination of the asymptotic behaviour of η shows that

$$\lim_{\|\boldsymbol{d}\| \to \infty} \eta = \frac{\sum_i w_i}{\sum_i \lambda_i}, \tag{A.6}$$

$$\lim_{\|\boldsymbol{d}\| \to 0} \eta = \frac{\sum_i w_i}{\sum_i \lambda_i}. \tag{A.7}$$

An extremum of η is reached for $\|\boldsymbol{d}\| = 1$, and is given by:

$$\eta_x = \frac{\sum_i w_i (1 - \cos(b_i))}{\sum_i \lambda_i (1 - \cos(a_i))}. \tag{A.8}$$

Note that this extremum value of η may be close to the asymptotic values (eqs. (A.6) and (A.7)), provided the angles between $\boldsymbol{d}$ and components of each basis vector are not very different. In general, function $\eta(\|\boldsymbol{d}\|)$ is not changing significantly with $\|\boldsymbol{d}\|$. Also, it is very unlikely that the value of $\|\boldsymbol{d}\|$ is exactly equal to one. Then, it is safe to use only one value for the parameter η, probably close to its asymptotic value. Of course, eigenvalues of the Hessian and weight matrix depend on a particular case. However, we are dealing with a sum of eigenvalues over all points, and it seems reasonable to assume that its value does not change significantly from case to case. Our results, obtained for two different meteorological situations, indirectly confirm that assumption.

REFERENCES

Axelsson, O. and Barker, V. A. 1984. *Finite-element solution of boundary-value problems. Theory and computation.* Academic Press (Computer Science and Applied Mathematics Series), 432 pp.

Chao, W. C. and Chang, L. P. 1992. Development of a four-dimensional variational assimilation system using the adjoint method at GLA. Part I: Dynamics. *Mon. Wea. Rev.* **120**, 1661–1673.

Conway, J. B. 1985. *A course in functional analysis.* Springer-Verlag, New York, 404 pp.

Courtier, P., Thepaut, J.-N. and Hollingsworth, A. 1992. A strategy for operational implementation of 4D-VAR. Proceedings of the ECMWF workshop on *Variational assimilation with emphasis on three-dimensional aspects.* Reading, 9–12 November 1992.

Derber, J. C. 1989. A variational continuous assimilation technique. *Mon. Wea. Rev.* **117**, 2437–2446.

DiMego, G. J. 1988. The National Meteorological Center Regional Analysis System. *Mon. Wea. Rev.* **116**, 977–1000.

Fletcher, R. 1972. *Conjugate gradient methods for unconstrained optimization*, ed. W. Murray, Academic Press, 144 pp.

Forsythe, G. E. and Strauss, E. G. 1955. On best conditioned matrices. Proceedings of the *Amer. Math. Soc.* **6**, 340–345.

Gill, P. E., Murray, W. and Wright, M. H. 1981. *Practical optimization.* Academic Press, London, 401 pp.

Golub, G. H. and Van Loan, C. F. 1989. *Matrix computations.* The John Hopkins University Press, Baltimore and London, 642 pp.

Hoffman, R. N. 1986. A four-dimensional analysis exactly satisfying equations of motion. *Mon. Wea. Rev.* **114**, 388–397.

Hoffman, K. and Kunze, R. 1971. *Linear algebra.* Prentice-Hall Inc., Second Edition, 406 pp.

Janjic, Z. I. 1990. The step-mountain coordinate: Physical package. *Mon. Wea. Rev.* **118**, 1429–1443.

Liu, D. C. and Nocedal, J. 1989. On the limited memory BFGS method for large scale optimization. *Math. Program.* **45**, 503–528.

Lorenc, A. 1986. Analysis methods for numerical weather prediction. *Quart. J. R. Met. Soc.* **112**, 1177–1194.

Luenberger, D. L. 1984. Linear and non-linear programming. Addison-Wesley, 2nd edition, 491 pp.

Mesinger, F., Janjic, Z. I., Nickovic, S., Gavrilov, D. and Deaven, D. 1988. The step mountain coordinate: model description and performance for cases of Alpine lee cyclogenesis and for a case of an Appalachian redevelopment. *Mon. Wea. Rev.* **116**, 1493–1518.

Nash, S. G. 1985. Preconditioning of truncated-Newton methods. *SIAM J. Sci. Stat. Comput.* **6**, 599–616.

Navon, I. M. and de Villiers, R. 1983. Combined penalty muliplier optimization methods to enforce integral invariant conservation. *Mon. Wea. Rev.* **111**, 1228–1243.

Navon, I. M. and Legler, D. M. 1987. Conjugate-gradient methods for large scale minimization in meteorology. *Mon. Wea. Rev.* **115**, 1479–1502.

Navon, I. M., Zou, X., Derber, J. C. and Sela, J. 1992. Variational data assimilation with an adiabatic version of the NMC spectral model. *Mon. Wea. Rev.* **120**, 1433–1446.

Nocedal, J. 1980. Updating quasi-Newton matrices with limited storage. *Math. of Computation* **35**, 773–782.

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

Shanno, D. F. 1978. Conjugate gradient methods with inexact line searches. *Math. Operations Res.* **3**, 244–256.

Shanno, D. F. 1985. Globally convergent conjugate gradient algorithms. *Math. Program.* **33**, 61–67.

Thacker, W. C. 1989. The role of the Hessian matrix in fitting models to measurements. *J. Geophys. Res.* **94**, C5, 6177–6196.

Thepaut, J.-N. and Courtier, P. 1991. Four-dimensional variational data assimilation using the adjoint of a multilevel primitive equation model. *Quart. J. R. Met. Soc.* **117**, 1225–1254.

Thepaut, J.-N., Vasiljevic, D., Courtier, P. and Pailleux, J. 1993. Variational assimilation of conventional meteorological observations with a multilevel primitive equation model. *Quart. J. R. Met. Soc.* **119**, 153–186.

Wang, Z. and Navon, I. M. 1992. *The adjoint truncated Newton algorithm for large-scale unconstrained optimization.* Report FSU-SCRI-92-170, The Florida State University, Tallahassee, Florida, USA.

Wang, Z., Navon, I. M., Le Dimet, F. X. and Zou, X. 1993. The second order adjoint analysis: theory and applications. *J. Atmos. Phys.*, in press.

Županski, M. 1993. Regional four-dimensional variational data assimilation in a quasi-operational forecasting environment. *Mon. Wea. Rev.* **121**, 2396–2408.