Variational Data Assimilation with an Adiabatic Version of the NMC Spectral Model*

I. M. NAVON
Department of Mathematics and Supercomputer Computations Research Institute, Florida State University, Tallahassee, Florida

X. ZOU
Supercomputer Computations Research Institute, Florida State University, Tallahassee, Florida

J. DERBER AND J. SELA
National Oceanic and Atmospheric Administration, National Meteorological Center, World Weather Building, Washington, D.C.

(Manuscript received 29 July 1991, in final form 10 October 1991)

ABSTRACT

Variational four-dimensional (4D) data assimilation is performed using an adiabatic version of the National Meteorological Center (NMC) baroclinic spectral primitive equation model with operationally analyzed fields as well as simulated datasets. Two limited-memory quasi-Newton minimization techniques were used to iteratively find the minimum of a cost function, with the NMC forecast as a constraint. The cost function consists of a weighted square sum of the differences between the model forecast and observations over a time interval. In all the experiments described in this paper, observations are available for all degrees of freedom of the model. The derivation of the adjoint of the discretized adiabatic NMC spectral model is presented. The creation of this adjoint model allows the gradient of the cost function with respect to the initial conditions to be computed using a single backward-in-time integration of the adjoint equations.

As an initial evaluation of the variational data-assimilation procedure, an assimilation system with a low-resolution version of the NMC spectral model was tested using fields from a Rossby–Haurwitz-wave solution as observations. The results were encouraging, with a significant reduction in the magnitudes of both the cost function and the norm of its gradient during the minimization process. In particular, the high-frequency noise exhibited in the rms of the divergence field, produced by random perturbation in the initial conditions, is largely eliminated after the variational data assimilation.

The performance of the assimilation scheme was examined in a more realistic configuration using the adiabatic NMC spectral model truncated at T40. Both operationally analyzed observations, consisting of vorticity, divergence, temperature, surface pressure and moisture fields (distributed at two time levels separated by a 6-h time interval), and model-generated data were variationally assimilated. The effect of the number of observation fields in time on the convergence rate of the minimization and the impacts due to the inclusion of the horizontal diffusion and the surface drag in the model and its adjoint on the convergence rate and the accuracy of the retrieval are addressed.

1. Introduction

Numerical weather prediction (NWP) is based on the integration of a dynamic system of partial differential equations modeling the behavior of the atmosphere. Therefore, discrete initial conditions describing the state of the atmosphere have to be provided prior to the integration, since they, along with the model equations, control the evolution of the solution trajectory in space and time.

In the last few years several important developments have taken place in NWP directed mainly in two different directions:

(a) The advent of more powerful computers has allowed the development of higher-resolution models including an ever-increasing number of physical processes and parameterizations of subgrid phenomena.

(b) New sources of data, such as satellite data, radar, profilers, and other remote-sensing devices, have become available. These data characteristically have a heterogeneous distribution in space and time as well as a nonrandom error structure.

Because of these advances in modeling and the observation networks, a need for an improved assimilation system has developed. There are many techniques used in data assimilation (Ghil et al. 1981; Ghil and Malanotte-Rizzoli 1991). Recently, considerable attention has been focused on variational methods for four-dimensional (4D) data assimilation (LeDimet and Talagrand 1986; Derber 1985; Lewis and Derber 1985; Hoffman 1986; Talagrand and Courtier 1987; Courtier and Talagrand 1987). This technique not only has

* G.F.D.I. Contribution No. 335.

Corresponding author address: Dr. I. M. Navon, Department of Mathematics and Supercomputer Research Institute, Florida State University, Tallahassee, FL 32306-4052.

broad applications for the assimilation of meteorological and oceanographical observations but is also applicable to sensitivity analysis (e.g., Cacuci 1981; Hall and Cacuci 1982, 1983), as well as to parameter estimation (e.g., Smedstad and O'Brien 1991).

The objective of variational data assimilation is to determine the optimal solution of an NWP model by fitting the model dynamics to data over an interval of time, where the optimality is measured by a cost function that expresses the degree of discrepancy between the model and the data. A direct approach for finding the optimal solution, suggested by Hoffman (1986), was found to be impractical for operational NWP models implemented on present-day computers. The expense of the variational assimilation can be reduced by using the adjoint of the assimilating model to calculate all of the components of the gradient of the cost function with respect to the initial conditions by one time integration. The adjoint model arises from the theory of optimization and optimal control of partial differential equations (developed over the last 30 years by mathematicians) (see, e.g., Pontryagyn et al. 1962; Lions 1971, 1988; Glowinski 1984), and whose applications are progressively propagating in various fields.

The use of adjoint equations in meteorology was pioneered by Marchuk (1974, 1982) and Penenko and Obraztsov (1976) and described by Kontarev (1980) and LeDimet and Talagrand (1986). Hall and Cacuci (1983) used the method to study the sensitivity of numerical models with respect to physical parameters. Courtier (1984) presented an adjoint variational data-assimilation method for the shallow-water equations model. Lewis and Derber (1985) used the adjoint method to solve a variational adjustment problem with advective constraints, while Derber (1985) used the adjoint method for a variational 4D data assimilation using the equations of a quasigeostrophic model as constraints. LeDimet and Talagrand (1986) used the method for data assimilation with a shallow-water equations model in all cases. The gradient of the cost function is computed by first integrating the forecast model forward and then by integrating the adjoint-model equations backward in time. During the backward integration of the adjoint equations, forcing terms are added to the adjoint equations at times when observational data is available. This method appears to be promising, especially for large dimensional data-assimilation problems. The calculation of the gradient of the cost function with respect to the initial condition vector requires less than twice the computer time of the integration of the forecast model for the dynamics-only model used in this paper. For more nonlinear problems, such as those contained in some physical parameterizations, the computational expense will increase, but the adjoint equations still represent the most efficient technique for calculating the gradient of the cost function.

Once the values of cost function and its gradient are available, different minimization methods can be employed to find the optimal solution. Among the most useful methods for solving an unconstrained cost function J of N variables are: (i) Newton's method and variations of it (O'Leary 1982; Toint 1981); (ii) conjugate-gradient (C-G) methods (Fletcher 1987; Gill et al. 1981); (iii) quasi-Newton methods (Shanno 1978; Nocedal 1980; Griewank and Toint 1982; Buckley and LeNir 1983), and (iv) truncated Newton (TN) methods (Nash 1984; Schlick and Fogelson 1992). The Newton and quasi-Newton methods have quadratic rates of convergence and superlinear rates of convergence, respectively, but require storage of Hessian matrices of size ($N \times N$). For large meteorological models, it is not yet possible to retain the matrices in direct memory. The C-G algorithms require storage of only a few vectors of length N. They have, however, a slower convergence rate than quasi-Newton methods.

A compromise between the faster convergence of the quasi-Newton methods and the smaller memory requirements of the C-G methods are limited memory quasi-Newton techniques. In these techniques, the accuracy of the Hessian matrix approximated depends on the available memory. The amount of memory usage can be controlled by the user. These techniques can be viewed as either an extension of the C-G method or as an approximation to the general quasi-Newton methods.

In this paper, results will be presented for a variational assimilation system being developed for the NMC adiabatic spectral model. After a brief description of the NMC spectral model and its discretization in space and time, we define the cost function and derive the adjoint-model equations (section 2). The minimization algorithms used are discussed in section 3, including issues of stepsize search, scaling, weighting, and verification of the correctness of the gradient. Results of variational data-assimilation experiments are then presented in section 4, including the accuracy of the retrieval of the meteorological fields, the impact of observation frequency, and the inclusion of a horizontal diffusion term and surface drag. A summary, general conclusions, and some of the outstanding problems are discussed in section 5.

2. Fundamentals of variational data assimilation

a. Brief description of the model

The NMC spectral adiabatic model is based on the primitive equations formulated with a spectral discretization in the horizontal and an Arakawa quadratic conserving finite differencing in the vertical with 18 σ layers. In order to take advantage of the spectral technique in the horizontal, a vorticity and divergence rep-

resentation of the momentum equations is used to eliminate the difficulties associated with the spectral representation of vector quantities on a sphere. Along with a moderate time filter applied to all fields, a semi-implicit time-integration scheme is applied to the divergence, temperature, and surface pressure equations. The vorticity and moisture equations are integrated explicitly. For more details on the NMC model equations, see Sela (1980).

b. Cost function

The objective of variational 4D assimilation is to find the solution to the constraining NMC forecast model that will best fit a series of observational fields distributed over some space and time interval. One possible measure of the fit, the cost function J, consists of a weighted least-squares fit of the model forecast to the observations. Assuming that the observations are given by analyzed fields, J is then represented by

$$J[\mathbf{X}(t_0)] = \frac{1}{2}\sum_{r=0}^{R}[\mathbf{X}(t_r) - \mathbf{X}^{\text{obs}}(t_r)]^{\mathrm{T}} \times \mathbf{W}(t_r)[\mathbf{X}(t_r) - \mathbf{X}^{\text{obs}}(t_r)], \quad (2.1)$$

where $\mathbf{X}(t_r)$ is the $N[=M(4K+1)]$ component vector containing values of divergence, vorticity, temperature, surface pressure, and moisture over all Gaussian grid points on all levels at time t_r; M is the number of Gaussian grid points at each level; K is the number of vertical levels; R is the number of analyzed fields in the assimilation interval; t_r represents the time when an observation occurs in the assimilation period window; $\mathbf{X}^{\text{obs}}(t_r)$ is the N-component vector of the analyzed values of $\mathbf{X}$ over all Gaussian grid points on all vertical levels at time t_r; and $\mathbf{W}(t_r)$ is an $N \times N$ diagonal weighting matrix with $\mathbf{W}_D$, $\mathbf{W}_\zeta$, $\mathbf{W}_T$, $\mathbf{W}_{\ln p_s}$, and $\mathbf{W}_q$ as its diagonal submatrices of the weighting factors for divergence, vorticity, temperature, surface pressure, and moisture fields, respectively.

One should notice that the cost function is defined in physical space rather than in spectral space, despite the fact that a spectral model is being used. The reason is that the large variability in the spectral coefficients of the variables makes the proper scaling of the variables difficult. Bad scaling will in turn negatively impact on the convergence of the minimization process due to the ill-conditioning of the Hessian matrix [matrix of second derivatives of the cost function with respect to the $\mathbf{X}(t_0)$].

c. Derivation of the gradient of the cost function

To find the minimum of the cost function J, all efficient minimization algorithms require the calculation of the gradient of the cost function. This can be done through the use of the adjoint equations for the forecast model. In this subsection, the derivation of the adjoint equations for the discretized NMC spectral model is presented. The discrete operations in the forward model have unique corresponding discrete operations in the adjoint model. The following derivation parallels the coding of the model where the algebraic operations are carried out by computer instructions.

The NMC implicit spectral model constraints can be written in the following general form

$$\mathbf{X}(t_0 + \Delta t) = \mathbf{F}_1[\mathbf{X}(t_0)] + \mathbf{L}_1\mathbf{X}(t_0) \quad (2.2a)$$

$$\mathbf{X}(t + \Delta t) = \mathbf{F}[\mathbf{X}(t)] + \mathbf{L}(t)\mathbf{X}(t - \Delta t) + \mathbf{m}(t)\mathbf{X}(t), \quad \text{for} \quad t > t_0 + \Delta t, \quad (2.2b)$$

where $\mathbf{F}_1(x)$ and $\mathbf{F}(x)$ are nonlinear operators, and $\mathbf{L}_1$, $\mathbf{L}$, and $\mathbf{m}$ are linear operators (which may depend on t).

To calculate the gradient of the cost function $\nabla J[\mathbf{X}(t_0)]$ with respect to the initial condition $\mathbf{X}(t_0)$, J', the change in the cost function resulting from a small perturbation $\mathbf{X}'(t_0)$ about the initial conditions $\mathbf{X}(t_0)$ is defined as

$$J'[\mathbf{X}(t_0)] = J[\mathbf{X}(t_0) + \mathbf{X}'(t_0)] - J[\mathbf{X}(t_0)]. \quad (2.3)$$

In limit as $\|\mathbf{X}'\| \to 0$, J' is the directional derivative in the $\mathbf{X}'(t_0)$ direction and is given by

$$J'[\mathbf{X}(t_0)] = \{\nabla J[\mathbf{X}(t_0)]\}^{\mathrm{T}}\mathbf{X}'(t_0). \quad (2.4)$$

The function J' may also be expressed by using (2.3) and the definition (2.1) of J,

$$J'[\mathbf{X}(t_0)] = \sum_{r=0}^{R}\{\mathbf{W}(t_r)[\mathbf{X}(t_r) - \mathbf{X}^{\text{obs}}(t_r)]\}^{\mathrm{T}}\mathbf{X}'(t_r). \quad (2.5)$$

where $\mathbf{X}'(t_r)$ is the perturbation in the forecast resulting from the initial perturbation, $\mathbf{X}'(t_0)$. Equating (2.4) and (2.5) results in

$$\{\nabla J[\mathbf{X}(t_0)]\}^{\mathrm{T}}\mathbf{X}'(t_0) = \sum_{r=0}^{R}\{\mathbf{W}(t_r)[\mathbf{X}(t_r) - \mathbf{X}^{\text{obs}}(t_r)]\}^{\mathrm{T}}\mathbf{X}'(t_r). \quad (2.6)$$

It is clear that if $\mathbf{X}'(t_r)$ can be expressed as a function of $\mathbf{X}'(t_0)$, then the gradient of the cost function with respect to the initial conditions can be found.

To express $\mathbf{X}'(t_r)$ in terms of $\mathbf{X}'(t_0)$, the forecast model (2.2) is linearized about the current model solution $\mathbf{X}(t)$. The linearized version of the forecast model (the tangent linear model) can be written as

$$\mathbf{X}'(t_0 + \Delta t) = \frac{\partial \mathbf{F}_1[\mathbf{X}(t_0)]\mathbf{X}'(t_0)}{\partial \mathbf{X}} + \mathbf{L}_1\mathbf{X}'(t_0) \quad (2.7a)$$

$$\mathbf{X}'(t + \Delta t) = \frac{\partial \mathbf{F}[\mathbf{X}(t)]\mathbf{X}'(t)}{\partial \mathbf{X}} + \mathbf{L}\mathbf{X}'(t - \Delta t) + \mathbf{m}(t)\mathbf{X}'(t), \quad (2.7b)$$

which may then be rewritten as

$$\mathbf{X}'(t_0 + \Delta t) = \mathbf{D}_1(t_0)\mathbf{X}'(t_0), \tag{2.8a}$$

$$\mathbf{X}'(t + \Delta t) = \mathbf{D}(t)\mathbf{X}'(t) + \mathbf{L}(t)\mathbf{X}'(t - \Delta t). \tag{2.8b}$$

where the matrices $\mathbf{D}_1$ and $\mathbf{D}$ are operator matrices depending on the values of the model state $\mathbf{X}(t)$, as well as on the time t, and the matrix $\mathbf{L}$ on the time t only. At each time step, the time-varying model state $\mathbf{X}(t)$ was written out on a storage unit while integrating the nonlinear NMC model forward and was subsequently read by a direct access for use in the adjoint model integration in reverse order. In this way the computer time required to skip data is saved. Note that $\mathbf{D}_1$, $\mathbf{D}$, and $\mathbf{L}$ are assumed real matrices. If one wishes to use complex vectors and matrices, all transposes in the derivation should be replaced with complex conjugate transposes.

Now from (2.8) the expression $\mathbf{X}'(t_r)$ can be derived in terms of $\mathbf{X}'(t_0)$ as follows:

$$\begin{aligned}\mathbf{X}'(t_r) &= \mathbf{D}(t_r - \Delta t)\mathbf{X}'(t_r - \Delta t) + \mathbf{L}(t_r - \Delta t) \\ &\quad \times \mathbf{X}'(t_r - 2\Delta t) \\ &= \mathbf{D}(t_r - \Delta t)\mathbf{D}(t_r - 2\Delta t)\mathbf{X}'(t_r - 2\Delta t) \\ &\quad + [\mathbf{D}(t_r - \Delta t)\mathbf{L}(t_r - 2\Delta t) + \mathbf{L}(t_r - \Delta t) \\ &\quad \times \mathbf{D}(t_r - 3\Delta t)]\mathbf{X}'(t_r - 3\Delta t) \\ &\quad + \mathbf{L}(t_r - \Delta t)\mathbf{L}(t_r - 3\Delta t)\mathbf{X}'(t_r - 4\Delta t) \\ &= \cdots \\ &\quad \vdots \\ &= \mathbf{P}_r\mathbf{X}'(t_0),\end{aligned} \tag{2.9}$$

where $\mathbf{P}_r$ represents the result of applying all the operator matrices in the linear model to obtain $\mathbf{X}'(t_r)$ from $\mathbf{X}'(t_0)$.

Using the symbolic expression of the linear version of the forecast model (2.9), (2.6) becomes

$$\{\nabla J[\mathbf{X}(t_0)]\}^{\mathrm{T}}\mathbf{X}'(t_0) = \sum_{r=0}^{R} \{\mathbf{W}(t_r)[\mathbf{X}(t_r) - \mathbf{X}^{\mathrm{obs}}(t_r)]\}^{\mathrm{T}}\mathbf{P}_r\mathbf{X}'(t_0). \tag{2.10}$$

This implies that

$$\nabla J[\mathbf{X}(t_0)] = \sum_{r=0}^{R} \mathbf{P}_r^{\mathrm{T}}\mathbf{W}(t_r)[\mathbf{X}(t_r) - \mathbf{X}^{\mathrm{obs}}(t_r)]. \tag{2.11}$$

Defining the adjoint equations as

$$\hat{\mathbf{X}}^r(t_0) = \mathbf{P}_r^{\mathrm{T}}\hat{\mathbf{X}}(t_r), \quad r = 1, \cdots, R, \tag{2.12}$$

where $\hat{\ }$ represents an adjoint variable, if the adjoint variable $\hat{\mathbf{X}}(t)$ at time t_r is initialized as

$$\hat{\mathbf{X}}^r(t_r) = \mathbf{W}(t_r)[\mathbf{X}(t_r) - \mathbf{X}^{\mathrm{obs}}(t_r)], \tag{2.13}$$

we obtain from (2.13)

$$\hat{\mathbf{X}}^r(t_0) = \mathbf{P}_r^{\mathrm{T}}\mathbf{W}(t_r)[\mathbf{X}(t_r) - \mathbf{X}^{\mathrm{obs}}(t_r)]. \tag{2.14}$$

Combining the results of (2.11) and (2.14), we arrive at the following formula for calculating the gradient of the cost function $\nabla J[\mathbf{X}(t_0)]$ with respect to the initial condition $\mathbf{X}(t_0)$:

$$\nabla J[\mathbf{X}(t_0)] = \sum_{r=0}^{R} \hat{\mathbf{X}}^r(t_0). \tag{2.15}$$

Since (2.12) and (2.15) are linear, $\nabla J[\mathbf{X}(t_0)]$ may be obtained by integrating the adjoint model from t_R to t_0 with zero initial conditions for the adjoint variables at time t_R while the weighted differences,

$$\hat{\mathbf{X}}^r(t_r) = \mathbf{W}(t_r)[\mathbf{X}(t_r) - \mathbf{X}^{\mathrm{obs}}(t_r)],$$

are inserted whenever an analysis time t_r ($r = 0, 1, \cdots, R$) is reached. Thus, a single integration of the adjoint model can yield the value of $\nabla J[\mathbf{X}(t_0)]$, the value of the gradient of the cost function with respect to the initial conditions.

d. Construction of the adjoint equations model and its verification of correctness

From (2.12) we note that the adjoint model equations consist of the transpose of the linearized version of the model. Thus, the linearized NMC model is developed first. Each quadratic term results in two terms, with both terms containing the model state and the perturbation.

If we view the linear model as the result of the multiplication of a number of operator matrices:

$$\mathbf{P} = \mathbf{A}_1\mathbf{A}_2 \cdots \mathbf{A}_N, \tag{2.16}$$

where each matrix $\mathbf{A}_i$ ($i = 1, \cdots, N$) represents either a subroutine or a single DO loop, then the adjoint model can be viewed as a product of adjoint subproblems

$$\mathbf{P}^{\mathrm{T}} = \mathbf{A}_N^{\mathrm{T}}\mathbf{A}_{N-1}^{\mathrm{T}} \cdots \mathbf{A}_1^{\mathrm{T}}. \tag{2.17}$$

In this way, the discrete adjoint model can be directly obtained from the discrete linear model, which in turn is obtained from the NMC spectral model by linearization around a model state. This simplifies not only the complexity of constructing the adjoint model but also avoids the inconsistency generally arising from the derivation of the adjoint equations in analytic form followed by the discrete approximation. A simple example of the construction of the discrete adjoint code from the discrete linear code is provided in the Appendix.

The correctness of the adjoint can be checked in two ways. First, at any level of the code, the development of the discrete adjoint model can be checked by applying the following identity:

$$(\mathbf{AQ})^{\mathrm{T}}(\mathbf{AQ}) = \mathbf{Q}^{\mathrm{T}}[\mathbf{A}^{\mathrm{T}}(\mathbf{AQ})], \quad (2.18)$$

where **Q** represents the input of the original code, and **A** represents either a single DO loop or a subroutine. All routines of the adjoint of the adiabatic NMC model were subjected to this test. A second verification of the correctness of the gradient is described below. Let

$$J(\mathbf{X} + \alpha\mathbf{h}) = J(\mathbf{X}) + \alpha\mathbf{h}^{\mathrm{T}}\nabla J(\mathbf{X}) + O(\alpha^2) \quad (2.19)$$

be a Taylor expansion of the cost function. The term α is a small scalar, and **h** is a vector of unit length (such as $\mathbf{h} = \nabla J \| \nabla J \|^{-1}$). Rewriting the above formulas, a function of α can be defined as

$$\Phi(\alpha) = \frac{J(\mathbf{X} + \alpha\mathbf{h}) - J(\mathbf{X})}{\alpha\mathbf{h}^{\mathrm{T}}\nabla J(\mathbf{X})} = 1 + O(\alpha). \quad (2.20)$$

For values of α that are small but not too close to the machine zero, one should expect to obtain a value for $\Phi(\alpha)$ that is close to 1. The value of $\Phi(\alpha)$ is shown in Fig. 1 for the T40 version of the NMC model. It is clearly seen that, for α between 10^{-7} and 10^{-15}, a unit value of $\Phi(\alpha)$ is obtained.

3. Descent algorithms for large-scale unconstrained minimization problems

Since the model already taxes the capability of the largest available computers, and the optimization of the initial conditions must conform to the operational requirements of timeliness, the choice of a robust and efficient minimization algorithm is crucial. Navon and Legler (1987) compared different C-G algorithms. Their conclusion was that the most robust and best-performing method was the Shanno and Phua (1980) quasi-Newton limited-memory algorithm. Liu and Nocedal (1988) compared the combined C-G quasi-Newton method of Buckley and LeNir (1983), the limited-memory quasi-Newton method described by Nocedal (1980) (called the L-BFGS method), and the partitioned quasi-Newton method of Griewank and Toint (1982). Zou et al. (1990) compared four of the state-of-the-art limited-memory quasi-Newton methods on several test problems, including problems in meteorology and oceanography. Their results show that the L-BFGS algorithm seems to be the most efficient and particularly robust. Since the Navon and Legler (1987) study, the truncated Newton algorithm has shown considerable promise (Nash and Nocedal 1989). Experiments with the variational 4D data assimilation on a shallow-water equations model with limited-memory quasi-Newton and truncated Newton minimization methods (Zou et al. 1990) showed that the truncated Newton algorithm might perform better than the limited-memory quasi-Newton method if a good preconditioner is found. The truncated Newton technique will be implemented on the system described in this paper.

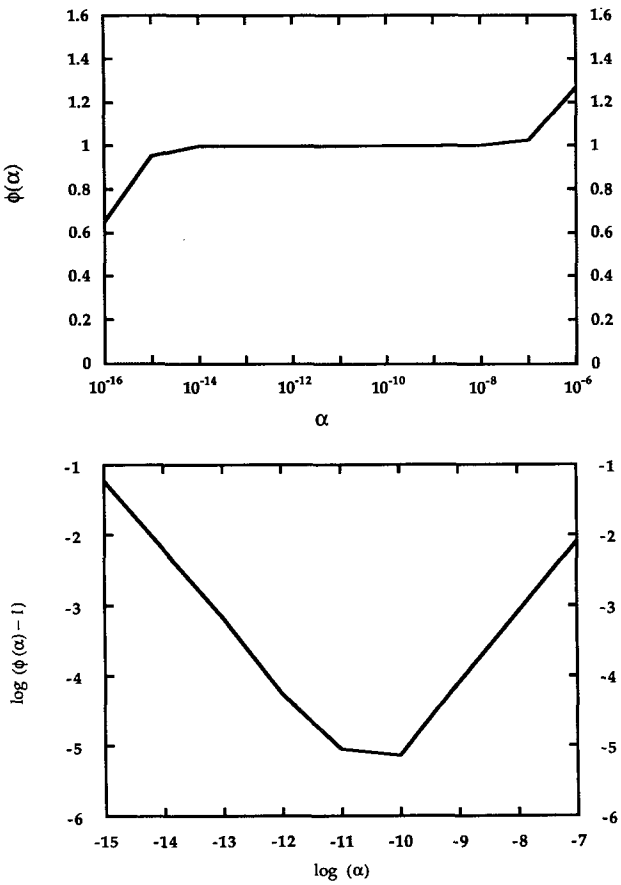

FIG. 1. Verification of the gradient calculation: (a) variation of Φ with respect to α; (b) variation of $\log(\Phi - 1)$ with respect to $\log\alpha$.

a. QN1: Limited-memory quasi-Newton method of Liu and Nocedal (1988)

For this study, the limited-memory quasi-Newton algorithm of Liu and Nocedal (1988) was chosen as one of the candidate minimization techniques since it deals with the critical issue of storage in large-scale problems. The QN1 update formula generates matrices using information from the last m quasi-Newton iterations, where m is the number of quasi-Newton updates determined by the user (generally $3 \leq m \leq 7$). After having used the m vector storage locations for m quasi-Newton updates, the quasi-Newton approximation of the Hessian matrix is updated by dropping the oldest information and replacing it by the newest information. A new search direction, which is an estimate of the relative change to the current variables vector that produces the maximum reduction in the cost function, is then computed. It employs a cubic line search required to satisfy a Wolfe (1969) condition, and a unit step size is always tried first. This algorithm uses a limited amount of storage, and the quasi-Newton approximation of the Hessian matrix is updated con-

tinuously. The general algorithm for QN1 can be found in the paper of Liu and Nocedal (1988).

b. QN2: Limited-memory quasi-Newton method of Shanno (1978) with step size from Derber (1985)

The QN2 method differs from the QN1 method in two aspects. First, the search direction is dependent only on information from the current and previous iterations [Shanno 1978, Eq. (20)]. Thus, the approximation of the Hessian may not be quite as accurate as that from QN1. This difference does not appear as important as the step-size calculations. The QN2 method assumes in the step-size calculation that the model solution resulting from a perturbation in the initial condition is well approximated by a model linearized around the unperturbed solution (Derber 1985). The optimal stepsize is the value for α_k, which minimizes the function $J[\mathbf{X}_k(t_0) + \alpha \mathbf{d}_k]$ or maximizes the functional reduction:

$$\Delta J(\mathbf{X}_k) = J[\mathbf{X}_k(t_0)] - J[\mathbf{X}_k(t_0) + \alpha \mathbf{d}_k], \quad (3.1)$$

where $\mathbf{d}_k$ is the search direction for iteration k, which has been defined already.

Define $\mathbf{X}'_g(t)$ as the difference between the forecasts resulting from integrating the initial condition $\mathbf{X}_k(t_0)$ and $\mathbf{X}_k(t_0) + \alpha_g \mathbf{d}_k$. The parameter α_g is a guess for the optimal step size based on the step size calculated from the previous iteration. Then, assuming that the forecast varies linearly with the initial perturbation results in the following estimate of the optimal step size:

$$\alpha_k = \frac{\alpha_g \sum_{r=0}^{R} \{\mathbf{W}(t_r)[\mathbf{X}_k(t_r) - \mathbf{X}_{\text{obs}}(t_r)]\}^{\mathrm{T}} \mathbf{X}'_g(t_r)}{\sum_{r=0}^{R} [\mathbf{W}(t_r)\mathbf{X}'_g(t_r)]^{\mathrm{T}} \mathbf{X}'_g(t_r)}. \quad (3.2)$$

Therefore, to calculate the optimal step size for one iteration, the forecast model needs to be integrated one time for the initial condition $\mathbf{X}_k(t_0) + \alpha_g \mathbf{d}_k$.

c. Scaling and weighting

Weights in the cost function serve a dual purpose: 1) they scale the cost function J to become a nondimensional quantity and 2) they reflect confidence in the quality of the observed data. Weights are usually chosen as the reciprocal of the variance of the observation errors. However, these variances are difficult to specify properly, and much more work is necessary in this area. Courtier and Talagrand (1990) also used a temporal weighting in which weights given to individual observations varied linearly with time, the total sum of the weights assuming the same value as the reciprocal of squared estimates of the statistical root-mean-square observational errors. The idea is that the model, not being perfect, cannot adjust uniformly to the whole set of observations at intermediate times. In the case of variational assimilation, where one performs a forecast starting from the final time of the assimilation period, a better adjustment to later observations is obviously preferable, and one may assign larger weights to more recent observations in the definition of the cost function.

In this experiment with Rossby–Haurwitz-wave observations, a constant diagonal weighting matrix with $\mathbf{W}_D = 10^7$, $\mathbf{W}_\zeta = 10^8$, $\mathbf{W}_T = 10^{-4}$, and $\mathbf{W}_Q = 2.5 \times 10^{-1}$ was employed. In the experiment with the T40 model, the weighting matrix is calculated by the following formula:

$$\mathbf{W}_\psi = \frac{1}{\max_{i,j,k} |\psi^{\text{obs}}_{i,j,k}(t_0) - \psi^{\text{obs}}_{i,j,k}(t_R)|^2}, \quad (3.3)$$

with similar expressions for the velocity potential, temperature, surface pressure, and moisture fields—that is, the inverse of the maximum difference between the two analyzed fields 6 h apart.

Scaling is a crucial issue in the success of nonlinear unconstrained optimization problems, and some research has been carried out on scaling nonlinear programming problems. It is well known that a badly scaled nonlinear programming problem can be almost impossible to solve (see also Navon and de Villiers 1983; Courtier and Talagrand 1990). An effective automatic scaling procedure would ease these difficulties and could also render problems that are well scaled easier to solve by improving the condition number of their Hessian matrix (Thacker 1989).

In the meteorological problem considered here, the variables in the control vector have enormously different magnitudes varying over a range of eight orders of magnitude. Scaling by variable transformation converts the variables from units that reflect the physical nature of the problem to units that display desirable properties for the minimization process. The general form of a scaling procedure is

$$\mathbf{X} = \mathbf{S}\mathbf{X}^s, \quad (3.4a)$$

$$\mathbf{g}^s = \mathbf{S}\mathbf{g}, \quad (3.4b)$$

$$\mathbf{H}^s = \mathbf{S}\mathbf{H}\mathbf{S}, \quad (3.4c)$$

where $\mathbf{S}$ is a diagonal scaling matrix, $\mathbf{X}$ and $\mathbf{g}$ are the state variable and the gradient, respectively, while $\mathbf{H}$ is the Hessian matrix. In the experiment with Rossby–Haurwitz waves, the divergence, vorticity, temperature, and surface pressure fields are scaled by the factors 10^{-7}, 10^{-4}, 10^2, and 1 respectively. For the operationally analyzed data, the scaling constants for the different fields are calculated by

$$\mathbf{S}_{\psi ii} = \max_{i,j,k} |\psi^{\text{obs}}_{i,j,k}(t_0) - \psi^{\text{obs}}_{i,j,k}(t_R)|/2, \quad (3.5)$$

and similarly for χ, T, $\ln p_s$, and q.

For complicated functions, difficulties may be encountered in choosing suitable scaling factors. There is no general rule to determine the best scaling factors for all minimization problems, and good scaling is problem dependent. A basic rule is that the variables of the scaled problem should be of similar magnitude and of order unity because within optimization routines convergence tolerances and other criteria are necessarily based on an implicit definition of "small" and "large," and, thus, variables with widely varying orders of magnitude may cause difficulties for some minimization algorithms (Gill et al. 1981). One simple direct way to determine the scaling factor is to use the typical values for different fields (for instance, 10^{-5} can be used as the scaling factor of vorticity). Further improvement to the condition number can be obtained by a more sophisticated scaling (Gill et al. 1981).

4. Numerical results of the variational data assimilation

A series of experiments were performed using the variational assimilation algorithm. These experiments were not intended as a complete evaluation but rather were directed toward examining a few aspects of the assimilation system. It was of primary importance to ensure that the system was properly constructed and that the minimization could be performed using a reasonably high-resolution system.

a. The Haurwitz-wave case

A simple experiment was conducted to verify that all the components of the techniques are properly working using analysis created from a Rossby–Haurwitz-wave solution. The minimization starts from the initial condition, to which a random perturbation is added on the Gaussian grids. The advantage of such an experiment is that one knows the exact solution and the value of the cost function at the minimum.

In this experiment, a coarse-resolution T6 model was employed with four vertical levels, a model spectral state is then completely described by 1344 independent real parameters (without moisture). However, the minimization was performed on the Gaussian grid, and the dimension for the initial conditions control variable is thus 2600.

The initial streamfunction was obtained by the formula

$$\psi = -a^2\omega \sin\phi + a^2\kappa \cos^m\phi \cos m\lambda, \quad (4.1)$$

where $\omega = \kappa = \Omega/10$, $m = 4$ defines a Haurwitz wave with zonal wavenumber 4. By using this streamfunction, the initial conditions can be defined as follows:

$$\zeta = \nabla^2\psi$$

$$D = 0$$

$$T = 266.4\sqrt{2}$$

$$\ln p_s = \frac{C_1[A + B\cos(m\lambda) + C\cos(2m\lambda)]}{R\bar{T}} + \ln(100), \quad (4.2)$$

where

$$C_1 = \frac{a^2\Omega^2}{10}$$

$$A = 1.05\cos^2\phi + 0.025\cos^{2m}\phi \times \left[(m+1)\cos^2\phi + 2m^2 - m - 2 - \frac{2m^2}{\cos^2\phi}\right]$$

$$B = \frac{2.2\cos^m\phi}{(m+1)(m+2)} \times [m^2 + 2m + 2 - (m+1)^2\cos^2\phi]$$

$$C = 0.025\cos^{2m}\phi[(m+1)\cos^2\phi - m - 2]. \quad (4.3)$$

The "observations" consisted of the complete space-time history of the fields ζ, D, T, and $\ln p_s$ over a 3-h time interval with the time steps being equal to 0.5 h. The variational data assimilation is carried out over the same time period.

The "observations" having been produced by the model itself, the minimum value of J is zero. Starting the minimization from an atmosphere that consists of the observations, plus at most 10% of a random perturbation of the gridded fields, and using the QN1 algorithm for the unconstrained descent process, a significant reduction of the values of both the cost function and the norm of its gradient has been obtained, as shown in Fig. 2. It is noted that the relative reduction

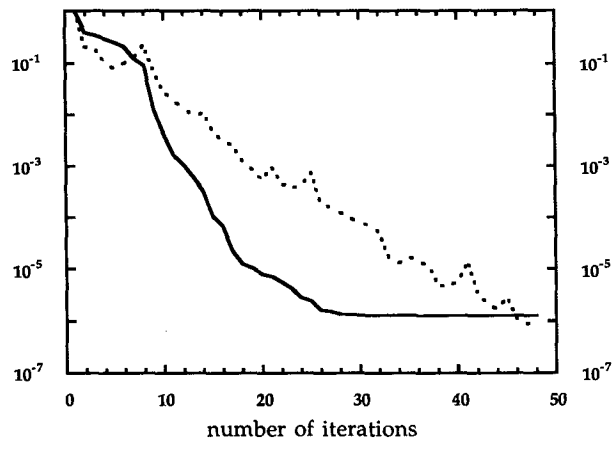

FIG. 2. Variations of the normalized cost function (J/J_0) (——) and normalized gradient ($\|\mathbf{g}\|\,\|\mathbf{g}_0\|^{-1}$) (· · ·) with the number of iterations using the QN1 method for model-generated Rossby–Haurwitz-wave observations.

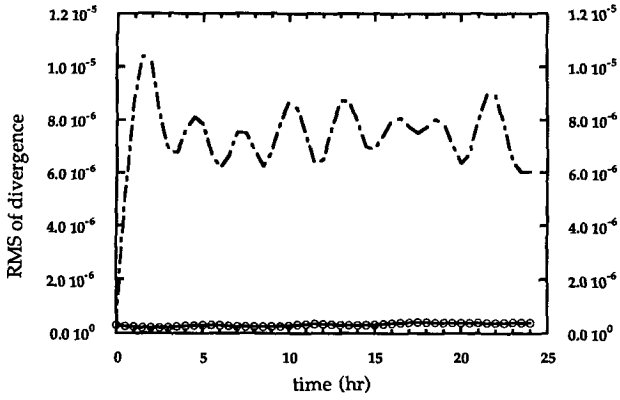

FIG. 3. Temporal variation, over a 24-h period, of the root-mean-square of the divergence field from the unperturbed initial conditions (——), perturbed initial conditions (– · –), and retrieved initial conditions (circles).

of the value of cost function J/J_0 is less than 10^{-5}. The norm of the gradient of the cost function was also reduced by five orders of magnitude. After about 30 iterations, the cost function reached its minimum.

The values of the difference fields between the retrieved and initial observations are much smaller than the differences between the random perturbation and initial observations and are almost equal to zero. Figure 3 shows the temporal variations of the rms of the divergence field, and the results indicate that the small oscillations for the Rossby–Haurwitz wave are successfully eliminated after the minimization.

b. T40 experiment with analyzed gridpoint data

In the second set of experiments, the observations consisted of the complete fields of vorticity, divergence, temperature, surface pressure, and moisture from the NMC global operational data-assimilation system 6 h apart. For this experiment, a truncation of T40 with 18 levels in the vertical is used. For the exact variable transformation from spectral space to physical grid space, or vice versa, a (90 × 46) Gaussian grid was used, and the dimension of the vector of control variables is 302 220.

The adjustment was performed on the 6-h interval $[t_0, t_R]$ preceding t_R. Experiments with vorticity and divergence used in the cost function converged slower than when streamfunction and velocity potential were used. Therefore, both the model solution and observations were changed to streamfunction and velocity potential when the transformation from spectral space to physical space, or vice versa, was carried out. Before trying to find the minimum of the cost function, it is important to realize that one cannot expect the minimum to be exactly zero. The decrease in the cost function itself is not a good measure of how close the solution is to the minimum. The decrease in the magnitude of the gradient of the cost function is a far better measure. Experience has shown that the decrease in the norm of the gradient is at most two or three orders of magnitude.

Using the QN1 and QN2 descent algorithms with the rough scaling defined in (3.5) and weighting matrix defined in (3.3), similar results are obtained in the reduction of the values of the cost function and the norm of its gradient from their initial values, as shown in Fig. 4. During the course of the first few iterations, the reduction in the value of the cost function is rather fast, and at about 18 iterations, the cost function seems to reach a minimum for both the QN1 and QN2 methods (solid and dotted lines in Fig. 4). The minimum value of the cost function was 18% of its original value. At 30 iterations, the norm of gradient decreased by one and one-half orders of the magnitude for both the min-

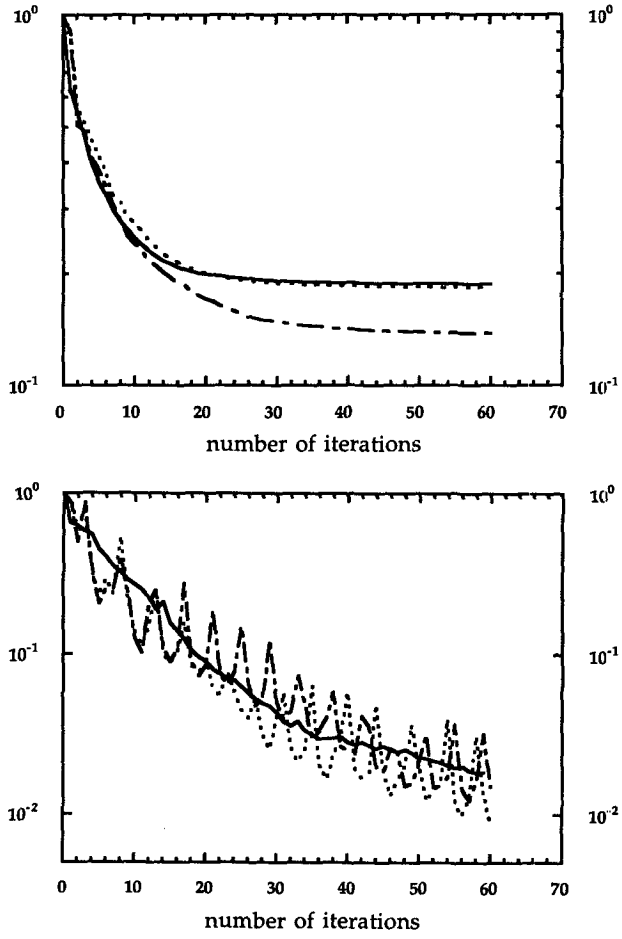

FIG. 4. Variations of the (a) normalized cost function (J/J_0) and (b) normalized gradient ($\|\mathbf{g}\|\,\|\mathbf{g}_0\|^{-1}$), with the number of iterations using the QN1 (· · ·) and QN2 (——) methods for operational analysis and the QN1 method in the case of normal-mode initialized analysis (– · –).

imization routines. After 30 iterations, the convergence rate becomes smaller for both of the QN1 and QN2 algorithms.

The two aforementioned minimization methods may be distinguished also by the number of function and gradient calls and by the resulting total CPU time and the smoothness with which the solution is approached. In 30 iterations, the QN2 method made 61 function calls and 30 gradient calls, while the QN1 method required 39 calls for both function and gradient evaluation. The additional function calls in the QN2 method increase the smoothness in the norm of the gradient because of a more accurate line-search algorithm. In terms of execution time, the QN2 method required 2041 CPU seconds, while the QN1 took 1827 CPU seconds on a CRAY-YMP supercomputer.

Since much of the lack of fit to the observations is in the small scale, the use of normal-mode-initialized observational data may improve the assimilation. The dash–dot line in Fig. 4 shows the results from the same experiment carried out with normal-mode-initialized data, using the QN1 minimization method. We see that the minimum of the cost function defined by the normal-mode-initialized observations is smaller than the one defined by the data without initialization. However, the convergence rate is the same for both cases. A similar conclusion was obtained using the QN2 method.

c. *Effect of the time density of observations on the convergence rate*

The effect of time density of observations on the convergence can be estimated by using model-generated observations. The experiments were carried out using one of the analyzed datasets as an initial condition

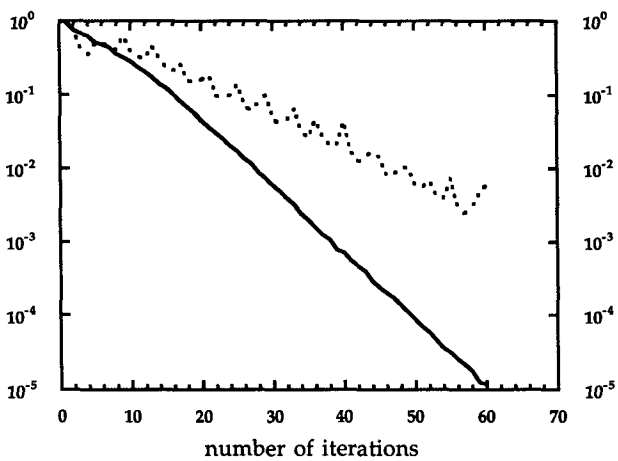

FIG. 5. Variations of the value of the normalized cost function (J/J_0) (——) and the normalized gradient $(\|g\| \|g_0\|^{-1})$ (· · ·), with the number of iterations using the QN1 method for model-generated observations.

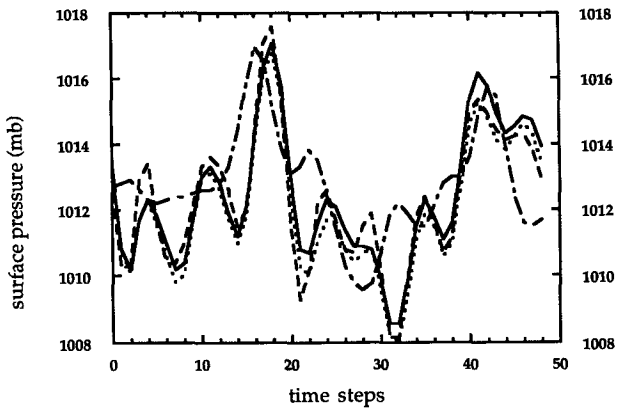

FIG. 6. Time variations of the surface pressure at a grid point from initial observations (——), the initial guess (– · –), retrieval after 14 iterations (– – –), 30 iterations (· · ·), and 60 iterations (——).

to generate observations in a 6-h window of assimilation and the initial conditions from an analysis time 6 h different as the initial guess for the QN1 minimization method. For these experiments it is known in advance that the cost function and the norm of its gradient should be zero at the minimum.

In order to better assess the results in section 4b, the similar experiment with model-generated observations (that is, one at the beginning and the other at the end of the 6-h assimilation window) was carried out. The results obtained for this case show that after 30 iterations the cost function decreased two and one-half orders of magnitude (solid line in Fig. 5). The norm of its gradient decreased about one and one-half orders of magnitude (dotted line in Fig. 5), which is similar to the results presented in Fig. 4b. If the minimization continues, the norm of gradient decreased another one order of magnitude after 60 iterations. The cost function also continues to decrease. However, the solution after 30 iterations is satisfactory, and the difference between the exact solution and the retrievals small. The small difference field after the minimization has decreased by at least one order of magnitude for all the fields.

Results of the minimization when observations are available at every time step in the assimilation window (in this case there are 13 time levels of observations) are similar to the results presented in Fig. 5. Again, the retrieved differences after 30 iterations are already quite satisfactory and of better quality. To illustrate the issue that the minimization procedure can be stopped prior to the cost function achieving its asymptotic rate of decrease, Fig. 6 presents the time integrations of the surface pressure at a grid point from the initial observations (solid line), the initial guess (dash–dot line) the retrieved initial conditions after 14 (dashed line), 30 (dotted line), and 60 (coincident with the solid line) iterations, respectively. We note that even after 14 it-

erations, when the cost function and the norm of the gradient have only decreased by less than one order of magnitude, most of the information has been recovered. After 30 iterations, the difference between the observations and retrieval becomes very small; that is, the corresponding integrations appear to have no phase discrepancy, and only small differences in amplitude can be discerned. After 60 iterations, the time variation is exactly the same. Thus, the variational data assimilation achieves most of the large-scale retrievals in the first 14–20 iterations, and, in the latter part of the minimization process, only small-scale features are being assimilated.

The effect of the length of the assimilation window on the convergence rate was also considered. Figure 7 presents the minimization performances when model-generated observations are available every time step between 0 and 1 h, 0 and 3 h, and 0 and 6 h, respectively. We see that, with the increase of the length of the window, the convergence rate decreases. This effect

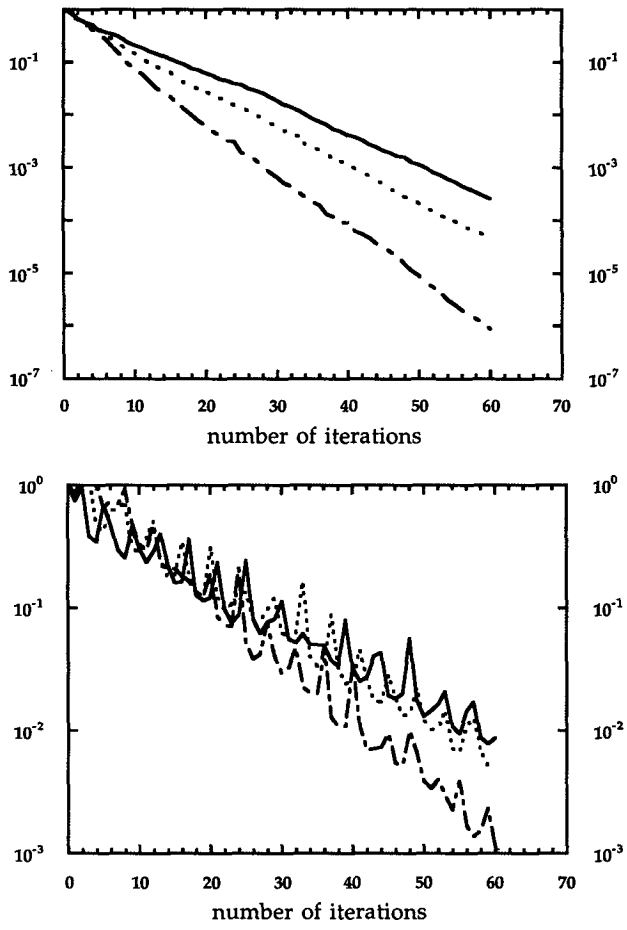

FIG. 7. Variation of the value of the (a) normalized cost function (J/J_0) and (b) normalized gradient ($\|\mathbf{g}\|\ \|\mathbf{g}_0\|^{-1}$) with the number of iterations using the QN1 method with model-generated observations with the operational analyzed data available every time step in 1-h (– · –), 3-h (· · ·), and 6-h (——) assimilation windows.

is due to the distortion of the spectrum of the Hessian matrix resulting in the worsening of the conditioning of the Hessian matrix.

These results indicate that, after a reduction of one or one and one-half orders of magnitude in the cost function and the norm of its gradient (as in the case of analyzed real-data assimilation), most of the large-scale features of the solution have already been recovered. The slower convergence rate of the minimization in the case of a longer assimilation window is due to the distortion of the Hessian by the model integration. Thus, the use of longer assimilation will require more computation not only from longer model integration but also from slower convergence.

d. Impacts of the horizontal diffusion and the surface drag

All of the results presented so far are obtained by using the adiabatic version of the NMC spectral model without any physics. Now, the impacts of the horizontal diffusion, the surface drag, and their combination on the ability of the variational data assimilation to recover the initial conditions is described.

The same experiment, as described in section 4c, with two time levels of observations, both at the beginning and the end of the 6-h assimilation window, was performed. The experiment in section 4c, without horizontal diffusion and the surface drag, is referred to as the reference experiment. Figure 8 shows the variation of the normalized cost function and norm of gradient with the number of iterations of the minimization process without the horizontal diffusion and the surface drag, with the horizontal diffusion ($\kappa_d = 8 \times 10^{15}$, $\kappa_T = 6 \times 10^{15}$, where κ_d is the diffusion coefficient for divergence and vorticity fields, and κ_T is the diffusion coefficient for temperature and moisture fields, respectively) and the very strong diffusion ($\kappa_d = 8 \times 10^{16}$ and $\kappa_T = 6 \times 10^{16}$). The experiment with very strong diffusion is presented by a dash–dot line, the reference experiment by a dashed line, and the experiment with the normal value of diffusion by a solid line. The convergence is faster with the horizontal-diffusion term present and much faster with the inclusion of a strong diffusion term. The norm of the gradient decreased by four orders of magnitude when a strong diffusion term was included, instead of only the two-orders-of-magnitude decrease obtained without inclusion of horizontal diffusion. Due to inclusion of the horizontal-diffusion term in the model, the forecast fields are smoother, and the minimization process proceeds faster toward the optimal solution.

If only the surface-drag term is added to both the adiabatic version of the model and the adjoint model, the minimization (see Fig. 8) performed better than the reference experiment. If both the surface-drag and horizontal-diffusion terms were added to the model, the minimization performed even better. It seems that

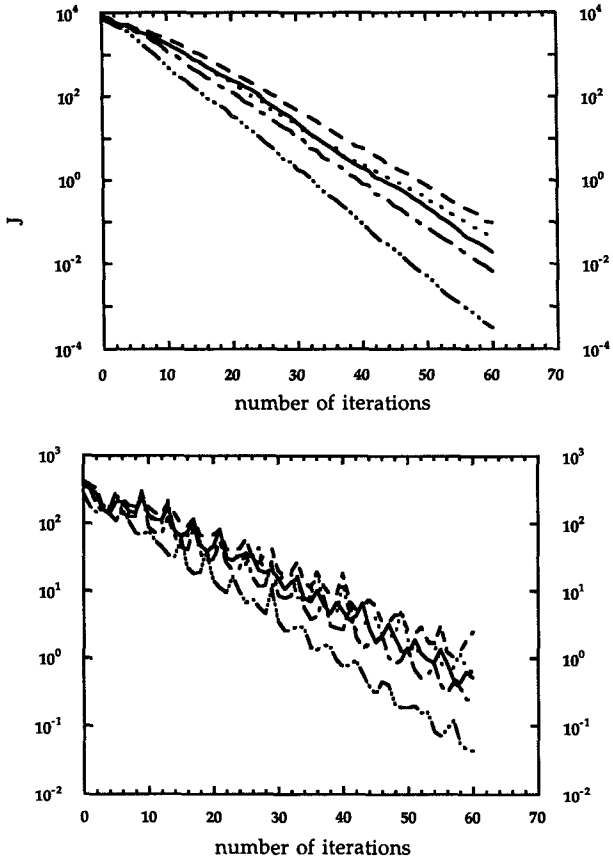

FIG. 8. Variations of the (a) cost function J and (b) gradient $\|\mathbf{g}\|$ with the number of iterations using the QN1 method without the horizontal diffusion and the surface drag (– – –), with only the horizontal diffusion (——), with only the surface drag (· · ·), with both the horizontal diffusion and the surface drag (– · –), and with only the strong horizontal diffusion (– · · · –).

both the horizontal diffusion and the surface drag were able to accelerate the convergence of minimization by controlling small-scale features present in the solution.

While the minimization algorithm performs well in all cases, the quality of the solution is dependent on the parameterization. To compare the accuracy, Fig. 9 illustrates the rms differences for the temperature and velocity potential between the assimilation solution after 30 iterations and the analysis at the end of 6 h as a function of model level. Figure 9a shows that the minimization without the surface drag and the horizontal diffusion gives a much more accurate retrieval in the middle levels of the model than on either the lower or the top levels. The maximum error occurs mainly near the bottom level for all of the fields of the model. There is also a large error near the top of the model for the velocity-potential field. However, these unbalanced errors in the solution were totally eliminated by including the strong horizontal-diffusion term in the model, which gives the best accuracy for the retrieval among all of the other experiments. The minimization with both the surface drag and the horizontal diffusion included in the model also yields similar results.

All of these results indicate that with the inclusion of some basic physics in the adiabatic version of the NMC spectral model the minimization converges faster and yields a more accurate solution to the variational data-assimilation problem. These results indicate that the inclusion of horizontal diffusion acts like a low-pass filter that smooths the small-scale physical features, thereby eliminating very small eigenvalues in the spectrum of the Hessian matrix of the cost function. This improves the condition number of the Hessian matrix and thus allows for a faster convergence rate of the descent process.

5. Summary and conclusions

In this paper, variational 4D data assimilation using the adjoint technique with a simplified version of the NMC spectral model has been presented. The best-fit trajectory is found using two limited-memory quasi-

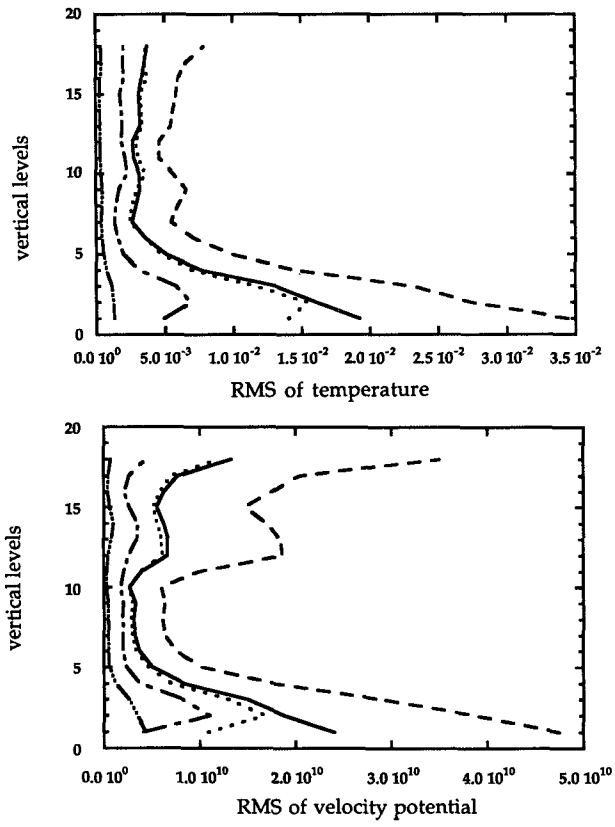

FIG. 9. Distributions of the rms differences for the (a) temperature field and (b) velocity-potential field between the minimization solution after 30 iterations and the operational analysis at the end of the 6-h assimilation interval. Results are shown for different physics: without the horizontal diffusion and the surface drag (– – –), with only the horizontal diffusion (——), with only the surface drag (· · ·), with both the horizontal diffusion and the surface drag (– · –), and with only the strong horizontal diffusion (– · · · –).

Newton nonlinear minimization algorithms. The minimization algorithms use the gradient of the cost function with respect to the initial conditions calculated using an adjoint technique described by LeDimet and Talagrand (1986). The variational assimilation was tested using a four-level T6 version of the adiabatic NMC primitive equation spectral model on a series of model-generated data from a Rossby–Haurwitz-wave initial condition. For this sample test problem, all of the components of the variational data assimilation performed well.

The assimilation technique performed well using NMC operationally analyzed data. In this case the assimilation technique used an 18-level T40 spectral model as a constraint. The weights and scaling factors were calculated from the analyzed data. A reasonable reduction in the cost function was achieved, and the quality of the retrieval (i.e., the reproduced initial conditions) is satisfactory. These experiments also show that the variational data assimilation was able to retrieve large-scale features of the fields during the first stage of the minimization procedure and that the convergence rate of the minimization depends very much on the number of available observations in time. Experiments with the horizontal-diffusion and surface-drag terms included in the model show that the addition of some physics to the model has beneficial effects on both the convergence rate and the accuracy of the retrieved initial conditions.

Perhaps the most significant result of this work is the ability to perform 4D data-assimilation experiments with a meteorologically realistic 18-level T40 spectral primitive equation model on present day computers. Further work on improving the convergence rate, the control of gravity-wave noise, and the addition of additional physics to the model is necessary.

Acknowledgments. The authors gratefully acknowledge the interest and advice of Prof. F. X. LeDimet in this work, the help during the initial stages of this research of Dr. K. Johnson of the Supercomputer Computations Research Institute (SCRI) and Florida State University, as well as the support of Drs. Pamela Stephens and Joseph Pandolfo, directors of GARP Division of the Atmospheric Sciences at the National Science Foundation (NSF). This research was funded by NSF Grant ATM-8806553. Additional support was provided by SCRI at Florida State University, which is partially funded by the Department of Energy through Contract DE-FC0583ER250000.

APPENDIX

Simple Example for Constructing the Discrete Adjoint Model

A simple example of the construction of an adjoint model from a linear model can be illustrated as follows. Consider a DO loop in the linear model,

$$\text{DO } 130\ I = 1, \quad N - 1$$

$$130\ \mathbf{X}(I) = a\mathbf{Y}(I + 1),$$

where $\mathbf{X}$ and $\mathbf{Y}$ are N-dimensional vectors. This DO loop is equivalent to the following two algebraic matrix equations

$$\begin{bmatrix} \mathbf{X}(1) \\ \mathbf{X}(2) \\ \cdot \\ \cdot \\ \cdot \\ \mathbf{X}(N-1) \end{bmatrix} = \begin{bmatrix} 0 & a & 0 & \cdots & 0 & 0 \\ 0 & 0 & a & \cdots & 0 & 0 \\ \cdot & \cdot & \cdot & \cdots & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdots & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdots & \cdot & \cdot \\ 0 & 0 & 0 & \cdots & 0 & a \end{bmatrix}_{(N-1)\times N} \begin{bmatrix} \mathbf{Y}(1) \\ \mathbf{Y}(2) \\ \cdot \\ \cdot \\ \cdot \\ \mathbf{Y}(N) \end{bmatrix}, \tag{A.1}$$

if $\mathbf{Y}(I)$ are not reused, or

$$\begin{bmatrix} \mathbf{Y}(1) \\ \mathbf{Y}(2) \\ \vdots \\ \\ \mathbf{Y}(N) \\ \mathbf{X}(1) \\ \mathbf{X}(2) \\ \vdots \\ \vdots \\ \mathbf{X}(N) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 1 \\ 0 & a & 0 & \cdots & 0 & 0 \\ 0 & 0 & a & \cdots & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 0 \end{bmatrix}_{2N\times N} \begin{bmatrix} \mathbf{Y}(1) \\ \mathbf{Y}(2) \\ \vdots \\ \vdots \\ \mathbf{Y}(N-1) \\ \mathbf{Y}(N) \end{bmatrix}, \tag{A.2}$$

if $\mathbf{Y}(I)$ are reused after the DO loop 130 in the linear model.

The adjoints of (A.1) and (A.2) can be written out directly by matrix transposition as

$$\begin{bmatrix} \mathbf{Y}(1) \\ \mathbf{Y}(2) \\ \cdot \\ \cdot \\ \cdot \\ \mathbf{Y}(N-1) \\ \mathbf{Y}(N) \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 & 0 \\ a & 0 & 0 & \cdots & 0 & 0 \\ 0 & a & 0 & \cdots & 0 & 0 \\ \cdot & \cdot & \cdot & \cdots & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdots & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdots & \cdot & \cdot \\ 0 & 0 & 0 & \cdots & 0 & a \end{bmatrix}_{N\times(N-1)} \begin{bmatrix} \mathbf{X}(1) \\ \mathbf{X}(2) \\ \cdot \\ \cdot \\ \cdot \\ \mathbf{X}(N-1) \\ \mathbf{X}(N) \end{bmatrix}, \tag{A.3}$$

or

$$\begin{bmatrix} \mathbf{Y}(1) \\ \mathbf{Y}(2) \\ \cdot \\ \cdot \\ \cdot \\ \mathbf{Y}(N) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 1 & 0 & \cdots & 0 & a & 0 & 0 & \cdots & 0 & 0 \\ \cdot & \cdot & \cdot & \cdots & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdots & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdots & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & 0 & \cdots & 1 & 0 & 0 & 0 & \cdots & a & 0 \end{bmatrix}_{N\times 2N} \begin{bmatrix} \mathbf{Y}(1) \\ \mathbf{Y}(2) \\ \cdot \\ \cdot \\ \cdot \\ \mathbf{X}(N) \end{bmatrix}, \tag{A.4}$$

respectively. Therefore the adjoint of the DO loop 130 will assume one of the following two forms

$$\begin{aligned} &\text{DO } 130\ I = 1,\quad N-1 \\ &130\ \mathbf{Y}(I+1) = a\mathbf{X}(I), \end{aligned} \tag{A.5}$$

or

$$\begin{aligned} &\text{DO } 140\ I = 1,\quad N-1 \\ &140\ \mathbf{Y}(I+1) = \mathbf{Y}(I+1) + a\mathbf{X}(I). \end{aligned} \tag{A.6}$$

REFERENCES

Buckley, A., and A. LeNir, 1983: QN-like variable storage conjugate gradients. *Math. Programming,* **27,** 155–175.

Cacuci, D. G., 1981: Sensitivity theory for nonlinear systems. I. Nonlinear functional analysis approach. *J. Math. Phys.,* **22,** 2794–2802.

Courtier, P., 1984: Présentation d'une méthode variationelle d'assimilation dynamique de dönnées meteorologique réparties dans l'espace et le temps. *Note E.E.R.M.,* **101,** 21.

——, and O. Talagrand, 1987: Variational assimilation of meteorological observations with the adjoint equation—Part I. Numerical results. *Quart. J. Roy. Meteor. Soc.,* **113,** 1329–1347.

——, and ——, 1990: Variational assimilation of meteorological observations with the direct and adjoint shallow-water equations. *Tellus,* **42A,** 531–549.

Derber J. C., 1985: The variational 4-D assimilation of analysis using filtered models as constraints. Ph.D. thesis, University of Wisconsin-Madison, 142 pp.

Fletcher, R., 1987: *Practical Methods of Optimization.* 2d ed., John Wiley & Sons, 436 pp.

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

——, S. Cohn, J. Tavantzis, K. Bube, and E. Isaacson, 1981: Applications of estimation theory to numerical weather prediction. *Dynamic Meteorology: Data Assimilation Methods,* L. Bengtsson, M. Ghil, and E. Kallen, Eds., Springer-Verlag, 330 pp.

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

Glowinski, R., 1984: *Numerical Methods for Nonlinear Variational Problems.* Springer-Verlag, 493 pp.

Griewank, A., and Ph. L. Toint, 1982: Partitioned variable metric updates for large structured optimization problems. *Numer. Math.,* **39,** 119–137.

Hall, M. C. G., and D. G. Cacuci, 1982: Sensitivity analysis of a radiative-convective model by the adjoint method. *J. Atmos. Sci.,* **39,** 2038–2050.

——, and ——, 1983: Physical interpretation of the adjoint functions for sensitivity analysis of atmospheric models. *J. Atmos. Sci.,* **40,** 2537–2546.

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

Kontarev, G., 1980: The adjoint equation technique applied to meteorological problems. ECMWF Tech. Rep., No. 21, Shinfield Park, Reading, RG2 9AX, U.K., 21 pp.

LeDimet, F. X., and O. Talagrand, 1986: Variational algorithms for analysis and assimilation of meteorological observations: Theoretical aspects. *Tellus,* **38A,** 97–110.

Lewis, J. M., and J. C. Derber, 1985: The use of adjoint equations to solve a variational adjustment problem with advective constrains. *Tellus,* **37A,** 309–322.

Lions, J. L., 1971: *Optimal Control of Systems Governed by Partial Differential Equations.* Springer-Verlag, 396 pp.

——, 1988: Exact controllability, stabilization and perturbation for distributed systems. *SIAM Rev.,* **30,** 1–68.

Liu, D. C., and J. Nocedal, 1988: On the limited memory BFGS method for large scale optimization. Tech. Rep. NAM 03, Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL, 26 pp.

Marchuk, G. I., 1974: Numerical solution of the problems of the dynamics of the atmosphere and ocean. *Leningrad, Gidrometeoizdat,* 303 pp.

——, 1982: Mathematical issues of industrial effluent optimization. *J. Meteor. Soc. Japan,* **60,** 481–485.

Nash, S. G., 1984: Newton-type minimization via the Lanczos method. *SIAM J. Numer. Anal.,* **21,** 770–788.

——, and J. Nocedal, 1989: A numerical study of the limited memory BFGS method and the truncated-Newton method for large scale optimization. Tech. Rep. NAM 02, Northwestern University, Evanston, IL 19 pp.

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

——, and D. M. Legler, 1987: Conjugate-gradient methods for large-

scale minimization in meteorology. *Mon. Wea. Rev.*, **115,** 1479–1502.

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

O'Leary, D. P., 1982: A discrete Newton algorithm for minimizing a function of many variables. *Math. Progr.*, **23,** 20–33.

Penenko, V., and N. N. Obraztsov, 1976: A variational initialization method for the fields of the meteorological elements. *Meteorol. Gidrol.*, **11,** 1–11.

Pontryagin, L. S., V. G. Boltyanskii, R. V. Gamkrelidze, and E. F. Mischenko, 1962: *The Mathematical Theory of Optimal Processes.* Interscience Publishers, 360 pp.

Schlick, T., and A. Fogelson, 1992: TNPACK—A truncated Newton minimization package for large-scale problems. *ACM Trans. Math. Software,* in press. [Available from T. Schlick, Courant Institute of Mathematical Sciences, 251 Mercer Street, New York, NY 10012.]

Sela, Joseph G., 1980: Spectral modeling at the National Meteorological Center. *Mon. Wea. Rev.*, **108,** 1279–1292.

Shanno, D. F., 1978: Conjugate gradient methods with inexact searches. *Meth. Oper. Res.*, **3,** 244–256.

——, and K. H. Phua, 1980: Remark on algorithm 500—A variable method subroutine for unconstrained nonlinear minimization. *ACM Trans. Math. Software,* **6,** 618–622.

Smedstad, O. M., and J. J. O'Brien, 1991: Variational data assimilation and parameter estimation in an equatorial Pacific Ocean model. *Prog. Oceanogr.*, **26,** 179–241.

Talagrand, O., and P. Courtier, 1987: Variational assimilation of meteorological observations with the adjoint vorticity equation—Part I. Theory. *Quart. J. Roy. Meteor. Soc.*, **113,** 1311–1328.

Thacker, W. C., 1989: The role of the Hessian matrix in fitting models to measurements. *J. Geophys. Res.*, **93,** 10 655–10 665.

Toint, Ph. L., 1981: Towards an efficient sparsity exploiting Newton method for minimization. *Sparse Matrices and Their Uses, The Institute of Mathematics and Its Applications Conference Series,* I. S. Duff, Ed., Academic Press, 57–87.

Wolfe, P., 1968: The secant method for simultaneous nonlinear equations. *Comm. Ass. Comput. Mach,* **2,** 12–13.

Zou X., I. M. Navon, F. X. LeDimet, A. Nouailler, and T. Schlick, 1990: A comparison of efficient large-scale minimization algorithms for optimal control applications in meteorology. SCRI Tech. Rep., FSU-SCRI-90-167, Tallahassee, FL, 25 pp.