Journal of Computational Physics **181,** 430–453 (2002)
doi:10.1006/jcph.2002.7136

On the Smoothness Constraints for Four-Dimensional Data Assimilation

Ching-Long Lin,* Tianfeng Chai,* and Juanzhen Sun†

**Department of Mechanical and Industrial Engineering and IIHR Hydroscience & Engineering, The University of Iowa, Iowa City, Iowa 52242-1527; and †National Center for Atmospheric Research,*
P.O. Box 3000, Boulder, Colorado 80307-3000
E-mail: ching-long-lin@uiowa.edu

Received March 23, 2001; revised October 15, 2001

An algorithm for determination of the weights of smoothness penalty constraints for the four-dimensional variational data assimilation technique is proposed and evaluated. To study the nature of smoothness penalty constraints, a simple nonlinear harmonic oscillator problem is first considered. Penalizing smoothness constraints is found to make the modified Hessian matrix of the cost function more positive definite, akin to the idea behind the modified line search Newton's methods. However, the use of the derivative smoothness constraints with a fixed coefficient does not warrant uniform imposition of these constraints at every iteration. A remedy is to control the ratio of the smoothness penalty function over the cost function, which can dramatically increase the positive definite area. On the other hand, the large smoothness coefficients obtained from this approach can deteriorate the convergence property of the minimization problem. Based on these observations, an algorithm for tuning the weights of smoothness constraints is proposed to overcome the aforementioned problems. The algorithm is first applied to a simple dynamic problem. It is then tested on the retrieval of microscale turbulent structures in a simulated convective boundary layer. This method is further evaluated on the retrieval of a strong meso-scale thunderstorm outflow from Doppler radar data. The results show that the algorithm yields efficient retrieval. © 2002 Elsevier Science (USA)

1. INTRODUCTION

The variational data assimilation method has been developed to combine limited observational data measured by remote sensing techniques, such as satellite and radar, with dynamical models to acquire more complete wind and temperature data for meteorological and oceanographic applications. For instance, Sun and Crook [11] used the four-dimensional variational data assimilation technique (4DVAR) to retrieve the wind and thermodynamic

0021-9991/02 $35.00

fields of a gust front from Doppler radar data. Sun and Crook [12, 13] further incorporated microphysical models into the 4DVAR model and retrieved the detailed wind, thermodynamics, and microphysics of a convective storm from radar data. Wu *et al.* [18] assimilated radar data of a severe thunderstorm into a cloud model using the 4DVAR technique. Lin *et al.* [3] applied the 4DVAR technique to retrieve microscale turbulent structures from a simulated convective boundary layer.

The concept of data assimilation is to find the controls that minimize the differences between the controls and observations subject to the constraints imposed by the model equations. Due to insufficient observational data and data error, data retrieval may be inaccurate or become ill-conditioned. Ooyama [8] indicated that derivative constraints can serve as a low-pass spatial filter to remove the undesirable errors at unresolved scales and ensure that the spatial scales of retrieved structures are not smaller than those of observations. Thacker [16] demonstrated that the use of smoothness penalties on the adjoint model of a three-wave simple dynamic system yields reasonable data retrieval in spite of sparse observations. The smoothness constraints can be regarded as bogus data, such as zero hypothetical slope and curvature in the objective space, reducing the ratio of observational data to the number of degrees of freedom of the model. Thus, these constraints can improve the conditioning of the minimization problem and speeds up the convergence. Long and Thacker [5] showed that with reduced observations in the data assimilation into an equatorial ocean model, penalizing departures of second derivatives of controls from smoothness results in satisfactory results. Sun *et al.* [10] demonstrated that the temporal and spatial smoothness constraints provide supplemental information on the retrieved variables and accordingly yield better solutions in the 4DVAR assimilation of simulated single-Doppler radar data. Sun and Crook [11] further applied the smoothness penalties to the adjoint retrieval of a gust front from the dataset observed during the Phoenix II experiment. The retrieval quality was found to improve remarkably. Yang and Xu [19] examined the effect of the spatial smoothness constraints on the systematic and nonsystematic errors in a one-dimensional advection equation. They found that these constraints can reduce both types of errors effectively. Lin *et al.* [3] demonstrated that spatial smoothness constraints can effectively improve the quality of microscale turbulent structures retrieved from a simulated convective planetary boundary layer.

It is noteworthy that the spatial smoothness penalty coefficients used in Long and Thacker [5], Sun and Crook [11], and Lin *et al.* [3] are 1.0, 0.05, and 0.00005, respectively. This indicates that the weights of smoothness penalty constraints for the mesoscale applications can be several orders of magnitude greater than those for the microscale applications. At times the weights for the same problem but with varying density of observational data require adjustment for optimal retrieval. It is desirable to have some rules to follow in determination of the weights of smoothness penalties. Therefore, the objectives of the paper are two-fold. One objective is to analyze a simple dynamic system to shed light on the nature of the smoothness penalty constraints and to establish some general guidelines in determination of the weights. The other objective is to use on these guidelines to develop a generic algorithm that can reduce the degrees of freedom in determination of the smoothness weights for large problems and that is applicable to the 4DVAR retrieval of atmospheric flow structures at both micro- and mesoscales. This feature is a necessity in developing an adaptive 4DVAR technique for integration of various sources of data without elaborate tuning of the weights.

The paper is organized as follows. In Section 2, a simple nonlinear harmonic oscillator is first considered to elucidate the features of smoothness constraints. In Section 3, we examine the condition numbers of the smoothness penalty function and the cost function. Their relationship with data density is investigated. The conditions for selecting suitable weights of smoothness constraints are discussed. Based on these conditions, an algorithm is proposed and tested on the simple problem. In Section 4, we briefly describe the 4DVAR technique used in this study. The formulation of the procedure for tuning smoothness constraints in 4DVAR based on the preceding algorithm is presented in Section 5. In Section 6, the proposed procedure is tested on the retrieval of microscale turbulent structures in a convective boundary layer using the approach of identical twin experiments. In these experiments, observational data are synthetically generated by the prediction model of the 4DVAR. In Section 7, the proposed procedure is applied to retrieve mesoscale convective atmospheric structures from Doppler radar data. Concluding remarks are made in Section 8.

2. A SIMPLE SYSTEM

Consider a simple harmonic oscillator, which describes the motion of a spring and is governed by the set of ordinary differential equations,

$$\frac{du}{dt} = -ax(1+bx^2) - cu, \tag{1}$$

$$\frac{dx}{dt} = u, \tag{2}$$

where u is the speed of the oscillator at displacement x. The values of parameters a, b, and c are arbitrarily set to $a=1$, $b=10$, and $c=1$. b controls the degree of nonlinearity and $b=0$ reduces to a linear system. The damping term $-cu$ represents a frictional force. A semi-implicit discretization of the above equations as suggested by [15] gives

$$\frac{1}{\Delta t}(u^{n+1} - u^n) = -\frac{a}{2}(x^{n+1} + x^n)\left[1 + \frac{b}{4}(x^{n+1} + x^n)^2\right] - \frac{c}{2}(u^{n+1} + u^n), \tag{3}$$

$$\frac{1}{\Delta t}(x^{n+1} - x^n) = \frac{1}{2}(u^{n+1} + u^n). \tag{4}$$

Here the superscript n denotes the time index and Δt the size of time step. With the specified initial conditions $u^0 = 0$, $x^0 = 1$, and the time step $\Delta t = 0.1$, we can advance the above discretized equations in time through Newton's method. Figure 1a shows the time histories of u^n and x^n from $n = 0$ to 50. If the damping coefficient c is set to zero, the amplitudes of u^n and x^n remain unchanged and the solution exhibits a periodic feature.

Assume that observational data x_{ob}^m are only available at every other m points. The data assimilation problem can be formulated as: Find the initial conditions (controls) u^0 and x^0 for Eqs. (3) and (4), whose solution best fits the observational data x_{ob}^m. For some cases presented later, we add random observational errors to the observational data: $x_{ob}^m = x_{ob}^m + \epsilon$, where $|\epsilon| \leq 0.1$. Thus, the maximum error amplitude is 10% of the prescribed initial condition $x^0 = 1$. The distribution of observational data x_{ob}^m containing the above error with $m = 2$ is displayed in Fig. 1a. The generic cost function J_o quantifying the difference between

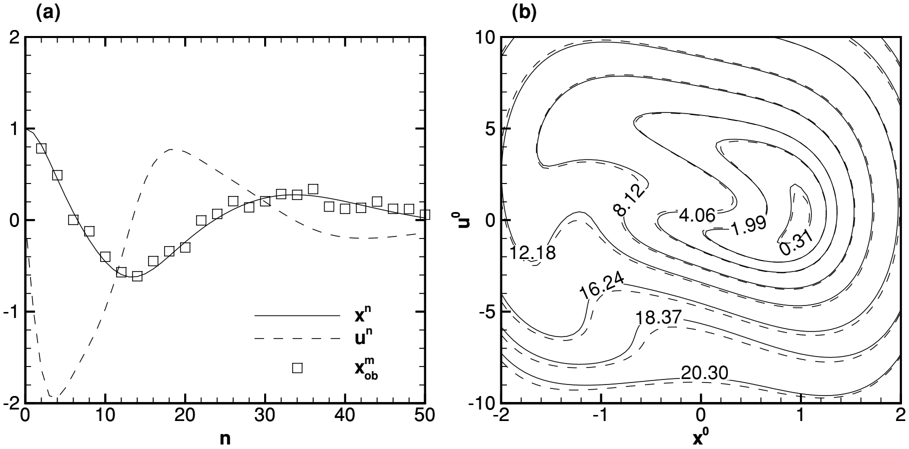

FIG. 1. (a) Time histories of x^n, u^n, and x^m_{ob} with random errors $|\epsilon| \leq 0.1$, (b) contours of the cost function J_o: without observational errors, solid lines; with observational errors $|\epsilon| \leq 0.1$, dashed lines.

observational data and model data is defined as

$$J_o = (\mathbf{x} - \mathbf{x}_{ob})^T \mathbf{O}^{-1} (\mathbf{x} - \mathbf{x}_{ob}), \tag{5}$$

where $\mathbf{x}$ stands for the solution vector and $\mathbf{x}_{ob}$ the observation vector. The superscripts T and -1 denote the transpose and inverse operations, respectively. Matrix $\mathbf{O}$ represents the observation error covariance matrix and is assumed a diagonal unity matrix so that errors are uncorrelated. Optimization of the controls is equivalent to minimization of the function J_o. The contours of the cost function Eq. (5) using the aforementioned observational data with $|\epsilon| = 0$ are shown in Fig. 1b. The optimal initial condition $(x^0, u^0) = (1, 0)$ is enclosed by crescent-shaped contours, which obscure the search of the optimal solution. As the contour level increases, the concave sides of these contours are found in the lower-left region of Fig. 1b. In the presence of observational errors $|\epsilon| \leq 0.1$, the contours of the cost function are slightly modified, but still look similar in shape to those without observational errors (solid contour lines versus dashed lines in Fig. 1b). With decreasing nonlinearity controlled by b in Eq. (1), these contours transform to an ellipse-like shape.

The addition of a smoothness penalty constraint P, based on the discrete second derivatives of x^n and u^n (Long and Thacker [5]), to Eq. (5) results in the modified cost function

$$J = J_o + P, \tag{6}$$

$$P = \gamma \sum_n S^n, \tag{7}$$

where

$$S^n = \begin{cases} [(x^1 - x^0)^2 + (u^1 - u^0)^2] & \text{if } n = 0, \\ [(x^{n+1} - 2x^n + x^{n-1})^2 + (u^{n+1} - 2u^n + u^{n-1})^2] & \text{if } n \geq 1, \end{cases}$$

and γ is the smoothness penalty coefficient. Figures 2a and 2b display the contours of P with $\gamma = 0.05$ and the contours of J with observational errors $|\epsilon| \leq 0.1$, respectively. Since the contours of the smoothness penalty function P exhibit an ellipse-like shape, the

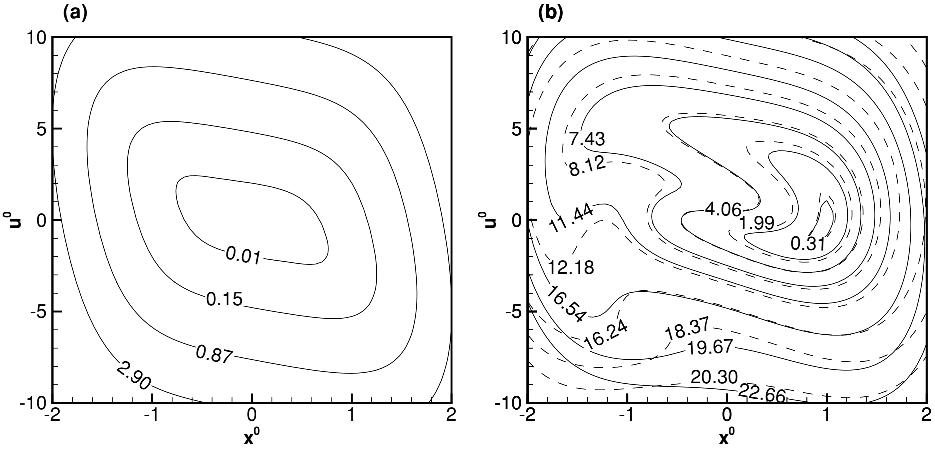

FIG. 2. (a) Contours of the smoothness penalty function P with $\gamma = 0.05$, (b) contours of: J, solid lines; J_o, dashed lines. $J = J_o + P$. Observational errors $|\epsilon| \leq 0.1$.

addition of P to J_o tends to reduce the concave contour curvature of the function J_o. For instance, compare the solid lines with the dashed lines in the vicinity of $(x^0, u^0) = (-1, -5)$ in Fig. 2b. Since the P contour level decreases inwards, the smoothness effect diminishes as the origin is approached.

Although Eq. (7) together with Eq. (6) resembles the quadratic penalty function in the quadratic penalty method, they are different in nature. That is, using the quadratic penalty method to find the minimizer of the cost function J_o subject to the constraints imposed by the two governing equations Eqs. (1) and (2), P in Eq. (6) shall be replaced by $\gamma \sum[\text{Eq. }(3)^2 + \text{Eq. }(4)^2]$, converting a constrained minimization problem to an unconstrained one. A larger γ means less constraint violation, but likely leading to an ill-conditioned problem. The quadratic penalty method is a weak constraint approach because the constraints are not strictly satisfied. In contract, the current method is a strong one in that the constraints are strictly satisfied by integrating the governing equations.

With a guess for initial conditions, we can compute the cost function J. By perturbing each of the input x^0 and u^0 in turn and then integrating Eqs. (3) and (4) to obtain the corresponding perturbation in J, the gradients $\partial J/\partial x^0$ and $\partial J/\partial u^0$ can be approximated by the second-order central difference method [1]. With these gradients, the limited-memory quasi-Newton BFGS [4] algorithm is applied to find an improved initial condition for the next iteration. The above steps are repeated until the convergence criterion is satisfied. Since the concave contours with large contour levels in Fig. 1b are located in the lower-left region, we choose $(x^0, u^0) = (-2, -10)$ as the first guess for the following numerical experiments to illustrate the effect of smoothness penalty constraints.

The results show that without observational errors the optimal solution $(x^0, u^0) = (1, 0)$ is recovered. With observational errors $|\epsilon| \leq 0.1$, the solution obtained at every iteration converges to an optimal solution slightly different from the exact one. If the observation error covariance matrix $\mathbf{O}$ in Eq. (5) is approximated, the above difference can be reduced. To illustrate the path taken by the minimization process without imposing any smoothness constraint, we mark in Fig. 3a with circles the locations of initial conditions retrieved at every iteration. The locus of these initial conditions approximately follows a clockwise route denoted by a dot-dashed curve to reach the optimal solution.

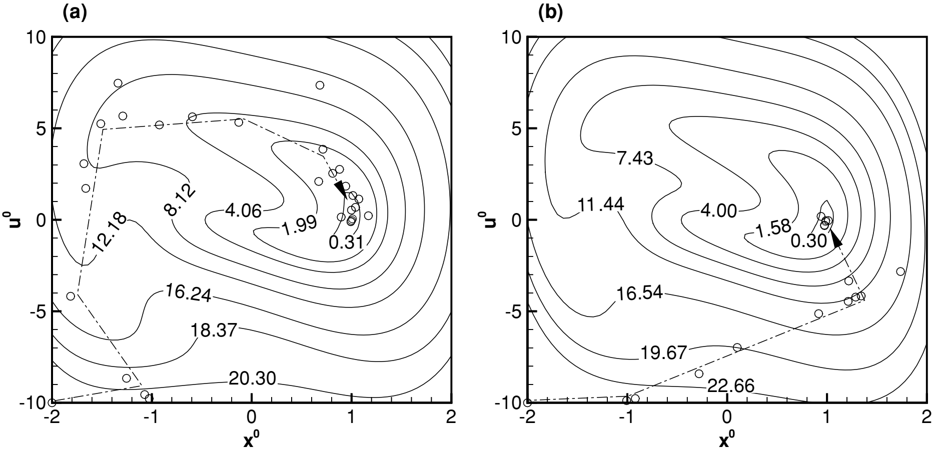

FIG. 3. Loci of initial conditions obtained at every iteration of the optimization process with the first guess $(x^0, u^0) = (-2, -10)$. (a) Without smoothness constraints P, (b) with smoothness constraints. Solid lines are contours of: (a) J_o, (b) J. $J = J_o + P$. Observational errors $|\epsilon| \leq 0.1$.

We next impose a smoothness constraint with a specified $\gamma = 0.05$, which is obtained by trial and error. The results show a significant reduction in number of iterations. For instance, with the convergence criterion

$$\left[\left(x_k^0 - x_{k-1}^0\right)^2 + \left(u_k^0 - u_{k-1}^0\right)^2\right]^{1/2} \leq 0.001, \tag{8}$$

where the subscript k denotes the iteration number, the case with $\gamma = 0.05$ takes 20 iterations to converge, whereas the case without the smoothness constraints takes 34 iterations. Figure 3b displays the contours of the cost function J for the case with $\gamma = 0.05$; notice that the curvatures in the lower-left region are greatly reduced. We shall examine the locus of the initial conditions retrieved at every iteration during this minimization. As shown in Fig. 3b, the modified contours allow the minimization to take a shorter path denoted by a dot-dashed curve to reach the optimal solution.

3. A GENERIC ALGORITHM

Although the concept of smoothness penalties is not new in the community of meteorology and oceanography as noted by Thacker [16], one may wonder whether a similar concept is put into practice in optimization in other science and engineering. A brief examination of the smoothness penalty terms in Eq. (7) reminds one of the standard algorithms for nonlinear constrained optimization, such as quadratic penalty and augmented Lagrangian methods. For the quadratic penalty method, the penalty coefficient adopted follows a sequence of increasing values to penalize gradually the constraint violations. The problem tends to become ill-conditioned as the penalty coefficients increase with iterations. Some eigenvalues of the Hessian matrix increase with the penalty coefficients, while some remain constant because the number of constraints is usually fewer than the rank of the original Hessian matrix [7]. So the augmented Lagrangian method is devised to overcome this problem. On the contrary, as pointed out by Sun and Crook [11], the smoothness penalty constraints in Eq. (7) shall not dominate in magnitude the original cost function to avoid excessive alteration of the minimizer. In this case, the penalty coefficients cannot be too large.

The pure Newton method converges rapidly if the Hessian matrix of the minimization problem is positive definite and the first guess is close to the minimizer. For practical applications, these conditions are unlikely to be satisfied, so the pure Newton method is often modified. One approach involves modification of the Hessian matrix of the cost function $\nabla^2 J_o$ to make it more positive definite during the minimization process [7]. The idea is to modify the eigenvalues of $\nabla^2 J_o$ by adding a sufficiently positive definite matrix $\mathbf{E}$ so that the modified Hessian matrix

$$\nabla^2 J = \nabla^2 J_o + \mathbf{E} \tag{9}$$

becomes more positive definite and better conditioned. In principle, $\mathbf{E}$ goes to zero at convergence. It can be proven via Zoutendijk's theorem that the global convergence of the modified line search Newton method follows if the condition numbers of the matrix $\nabla^2 J$ are bounded whenever those of $\nabla^2 J_o$ are bounded during the minimization process [7]. Mathematically, it is expressed as

$$C_k = \|\nabla^2 J_k\| \left\|\nabla^2 J_k^{-1}\right\| \leq D \text{ for all iterations } k, \tag{10}$$

where C_k is the condition number of the Hessian matrix $\nabla^2 J_k$ at iteration k, $\|\cdot\|$ denotes the norm of a matrix, and D is a real positive number. The simplest choice of the matrix $\mathbf{E}$ is $\omega\mathbf{I}$ where ω is a scalar and $\mathbf{I}$ is an identity matrix. Other algorithms along this line of thought include the modified Cholesky approach. The approach of adding matrix $\omega\mathbf{I}$ needs information on negative eigenvalues of $\nabla^2 J_o$, while the modified Cholesky method requires performing Cholesky factorization of the Hessian matrix, which is computationally expensive for large problems. Thus, application of these two modified Newton methods to the 4DVAR problems described in Sections 6 and 7 is not practical.

It is noted that solving a system of algebraic equations $\nabla^2 J_k p = -\nabla J_k$ in a minimization problem is to obtain a search direction p. Addition of the matrix $\mathbf{E}$ to the Hessian matrix can be regarded as a preconditioning technique to the minimization problem for improving the search direction. One should recognize that solving $\nabla^2 J_k p = -\nabla J_k$ using an efficient preconditioned iterative solver has nothing to do with improving the search direction. Besides, for large-scale minimization problems, different strategies and algorithms are developed to construct and store the Hessian matrix and solve the linear system more effectively [7].

In view of numerous successful examples of improving the conditioning of the data assimilation problems by imposing the smoothness constraints as reviewed in Section 1, it is speculated that the Hessian matrix of the smoothness penalty function $\nabla^2 P$ (Eq. (7)) plays the same role as matrix $\mathbf{E}$ in the modified Newton methods. For verification, Figures 4a and 4b display the diagonal entries of the Hessian matrix $\nabla^2 P$. The results show that they are positive everywhere and increase in magnitude as departing from the origin. Computation of the eigenvalues of the Hessian matrix $\nabla^2 P$ reveals that 96% of the domain is positive definite. In addition, the distributions of the condition number C of $\nabla^2 P$ are found to cluster around 8. Therefore, matrix $\nabla^2 P$ is sufficiently positive definite and is well-conditioned as required for matrix $\mathbf{E}$ in Eq. (9) for the modified Newton methods.

In order to investigate the effect of the modified Hessian matrix on the aforementioned simple dynamic problem, Figure 5 compares the regions associated with positive definite Hessian $\nabla^2 J_o$ with those that use a penalty coefficient $\gamma = 0.05$. Plotted are two different densities of data: $m = 2$ and 4 (note that data x_{ob}^m in Fig. 1a are available at every other

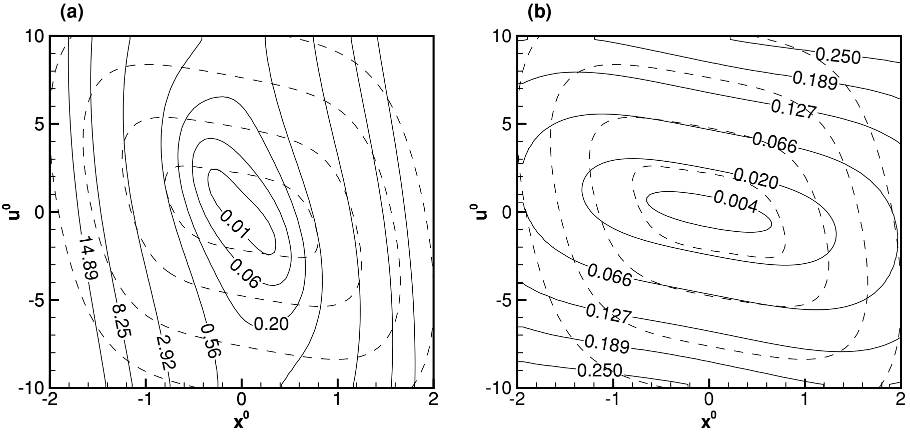

FIG. 4. Contours of diagonal entries of the Hessian matrix of the penalty term $\nabla^2 P$. Solid lines, (a) $\partial^2 P/\partial(x^0)^2$, (b) $\partial^2 P/\partial(u^0)^2$. Dashed lines represent contours of the smoothness penalty function P.

$m = 2$ point). In comparison with Fig. 5a for $m = 2$, the region of the positive definite $\nabla^2 J_o$ for $m = 4$ may become narrower or even disconnected as marked by letters A and B in Fig. 5b. By imposing smoothness penalty constraints, the positive definite area increases and the narrow positive definite region marked by A in Fig. 5b becomes much wider, suggesting improvement of the conditioning of the problem. In general, with reduced data density, the second-order information of the cost function is roughly preserved but distorted and reduces in magnitude. In contrast, the nature of the smoothness penalty function is independent of data density, and the contour levels of P and $\nabla^2 P$ are fixed as shown in Fig. 4 once the penalty coefficient γ is specified. Consequently, with decreasing data density, e.g., $m = 2, 4, 8$ with $\gamma = 0.05$, the percentages of the positive definite areas are 46%, 50%, and 54%, respectively, in an increasing order. The minimizer, however, drifts away from the actual one with increasing smoothness penalty.

To examine the degree of smoothness effect at every set of (x^0, u^0), we show in Fig. 6 the ratio R of the penalty function P with $\gamma = 0.05$ over the generic cost function J_o with

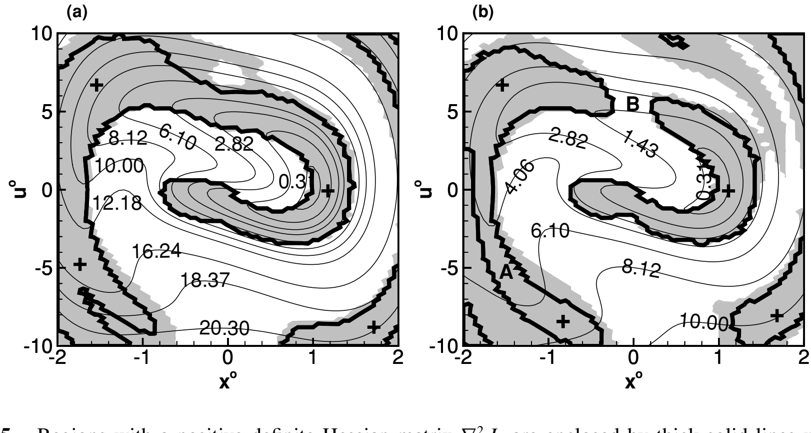

FIG. 5. Regions with a positive definite Hessian matrix $\nabla^2 J_o$ are enclosed by thick solid lines with "+" symbols. Gray color marks regions with a positive definite Hessian matrix $\nabla^2 J$ where $J = J_o + P$ and $\gamma = 0.05$. Thin solid lines denote contours of J_o. $m =$: (a) 2; (b) 4, where data are available at every other m points.

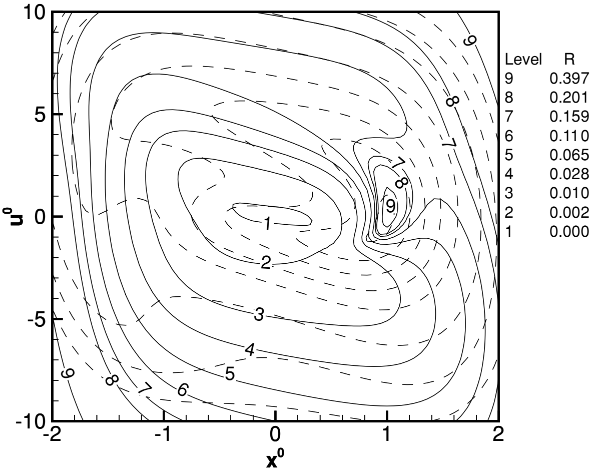

FIG. 6. Contours of the ratio $R = P/J_o$ with observational errors $|\epsilon| \leq 0.1$ and the penalty coefficient $\gamma = 0.05$, solid lines; contours of the function J with observational errors $|\epsilon| \leq 0.1$, dashed lines.

$|\epsilon| \leq 0.1$ and $m = 2$. Because of large R near the corners of Fig. 6 and near the optimal solution, e.g., contour level 9 with $R = 0.397$, we expect to see more effect of the smoothness penalty constraint in these regions. With $m = 4$, the contours of R look similar but with higher contour levels. This explains why the smoothness constraint is more effective near letter A in Fig. 5b than B. One strategy to apply smoothness constraints more uniformly is to impose a constant R, which maintains smoothness at a certain level relative to the underlying cost function throughout the minimization process. Based on this principle, the modified Hessian matrix $\nabla^2 J$ for $m = 2$ and 4 are reanalyzed by imposing $R = 0.5$. Now 91% and 82% of the domain in Figs. 7a and 7b are positive definite as compared with 46% and 50% in Figs. 5a and 5b which use $\gamma = 0.05$.

Evidently a constant R implies varying γ. Thus, in order to achieve uniform effect of smoothness constraints at every iteration k, a set of penalty coefficients γ_k shall be considered to replace a fixed γ. The concept of using a set of penalty coefficients is not

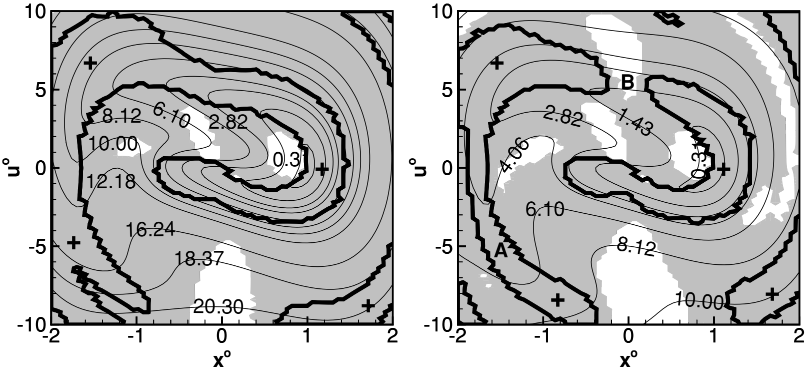

FIG. 7. Regions with a positive definite Hessian matrix $\nabla^2 J_o$ are enclosed by thick solid lines with "+" symbols. Gray color marks regions with a positive definite Hessian matrix $\nabla^2 J$ where $J = J_o + P$ and $R = P/J_o = 0.5$. Thin solid lines denote contours of J_o. $m =$: (a) 2; (b) 4, where data are available at every other m points.

new and is applied in the quadratic penalty method. However, as opposed to the quadratic penalty method in which γ_k increases with iterations to penalize constraint violations, γ_k in the smoothness penalty constraints, in principle, shall decrease with iterations to approach the actual minimizer as in the generic modified Newton's method and to satisfy the global convergence property Eq. (10). That is, if the problem is well-posed and γ_k decreases with iterations, condition (10) is likely to be met and the global convergence is guaranteed.

The use of a constant R, however, does not warrant decrease of γ_k with iterations and can deteriorate convergence. To overcome this problem, we propose the following algorithm in calculation of the smoothness penalty coefficient γ_k.

Algorithm 1

```
1. Initialization: $\gamma_0 = 1000$ (some large number), $R_\circ = 0.5$ (the upper limit on $R$)
2. Do loop $k = 0, 1, 2, \ldots$
3. Forward integration of the model
4. Calculation of $P$ and $J_o$
5. $R = P/J_o$
6. $\gamma_{k+1} = \gamma_k$
7. IF ($R > R_\circ$) THEN
8. $P = R_\circ/R \times P$
9. $\gamma_{k+1} = R_\circ/R \times \gamma_k$
10. END IF
11. Minimization
12. End do
```

$R_\circ$ sets a limit on the degree of smoothness constraints that can be applied relative to the cost function at iteration k. The algorithm ensures that the smoothness penalty function does not dominate the cost function and is controlled at a desired level locally, namely $R \leq R_\circ$. If $R > R_\circ$, γ_k has to be adjusted to relax smoothness penalty constraints. As a consequence, γ_k can only decrease with increasing k, which is critical in achieving global convergence. The selection of the optimal $R_\circ$ value depends on the nature of the dynamic system because the curvatures of the controls are problem-dependent. Based on the contours of the ratio R in Fig. 6, we choose $R_\circ = 0.5$ to test the algorithm on the simple system. With the same first guess $(x^0, u^0) = (-2, -10)$, it takes 21 and 37 iterations for $m = 2$ and 4 to satisfy the convergence criterion Eq. (8) and the retrieved minimizers are $(x^0, u^0) = (0.980, -0.244)$ and $(0.978, 0.113)$. As for $\gamma = 0.05$ cases, it takes 20 and 38 iterations for $m = 2$ and 4 to get $(x^0, u^0) = (0.974, -0.303)$ and $(0.956, -0.184)$. The above algorithm seems to produce slightly improved solutions. The initial γ_k value obtained from the above algorithm with $R_\circ = 0.5$ and $m = 2$ is 0.043, close to the value of 0.05 obtained by trial and error for the constant γ case. After 16 iterations, γ_k changes to 0.034 because the condition $R > R_\circ$ at Step 7 of Algorithm 1 is met.

In the following sections, the algorithm will be tested on large 4DVAR problems at different temporal and spatial scales. Unlike the above simple dynamic system, these problems have much more controls and the tuning of the penalty coefficients is rather time consuming. We shall demonstrate that the use of the above algorithm in these problems is able to effectively determine appropriate penalty coefficients at different scales with a single $R_\circ$.

4. FOUR-DIMENSIONAL DATA ASSIMILATION MODEL

First, we briefly describe the four-dimensional variational data assimilation technique (4DVAR) developed for the retrieval of atmospheric flow structures [3]. The technique is based upon the single-Doppler parameter retrieval system [10] and consists of two components: a prediction model and an optimization algorithm. The technique uses methods of control theory to find three-dimensional wind and temperature fields that best fit observational data.

4.1. Prediction Model

The 4DVAR technique first solves filtered incompressible Navier–Stokes equations with Boussinesq approximation, which are subject to a prescribed vertical mean temperature gradient,

$$\frac{\partial U_i}{\partial x_i} = 0 \tag{11}$$

$$\frac{\partial U_i}{\partial t} + \frac{\partial (U_j U_i)}{\partial x_j} = -\frac{1}{\rho_\circ}\frac{\partial P}{\partial x_i} + \delta_{i3}\frac{g\theta}{\Theta_\circ} + \nu\frac{\partial^2 U_i}{\partial x_j \partial x_j} \tag{12}$$

$$\frac{\partial \theta}{\partial t} + \frac{\partial (U_j \theta)}{\partial x_j} + U_3\frac{d\Theta}{dx_3} = \kappa\frac{\partial^2(\theta + \Theta)}{\partial x_j \partial x_j}, \tag{13}$$

where U_1, U_2, and U_3 (U, V, and W) are velocity components in the respective x_1, x_2, and x_3 (x, y, and z; east, north, and vertical) directions. θ, Θ, and $\Theta_\circ$ are fluctuating, background, and reference virtual potential temperature, respectively. Repeated indices imply summation. The eddy viscosity ν and thermal diffusivity κ are functions of height. The second-order finite volume method is applied for spatial differencing and the second-order Adam–Bashforth method is used for advancing dependent variables in time. The continuity equation is satisfied by solving a pressure-Poisson equation derived from Eqs. (11) and (12). The lateral boundary conditions for U, V, W, and θ fields at each time step are obtained through linear interpolation between observational data at different times. The gradient-free boundary condition is imposed at the top of the domain for U, V, and θ fields, whereas for W field the Dirichlet boundary condition $W = 0$ is used. If reflectivity data are available, an additional conservation equation for reflectivity Z shall be included in the numerical model.

4.2. Optimization Procedure

The optimization procedure involves the minimization of a cost function subject to the constraints imposed by the prediction model equations Eqs. (11)–(13). The generic cost function J_o is defined as

$$J_o = \sum_{0 \le n \le N} (\mathbf{H}\mathbf{x}^n - \mathbf{y}^n)^T \mathbf{O}^{-1} (\mathbf{H}\mathbf{x}^n - \mathbf{y}^n), \tag{14}$$

where the superscript n is the time index, $\mathbf{x}^n$ is the state vector, and $\mathbf{y}^n$ is the observation vector. Since $\mathbf{x}^n$ and $\mathbf{y}^n$ can be different variables on different grids, the observation operator $\mathbf{H}$ represents both a transformation between different grid meshes and an analytical function that relates model variables (i.e., U, V, and W) to observation variables (i.e., radial velocity

V_{rad}). **O** is the error covariance matrix that includes two sources of error: error in the observation vector $\mathbf{y}^n$ and error in the observation operator **H**. Like the cost function of the simple harmonic oscillator Eq. (5), we assume that the errors are uncorrelated so that the observation error covariance matrix **O** reduces to a diagonal matrix.

A three-dimensional velocity field can be converted to a radial velocity field through the relation,

$$V_{\text{rad}} = \frac{x - x_\circ}{r} U + \frac{y - y_\circ}{r} V + \frac{z - z_\circ}{r} (W - V_T), \tag{15}$$

where $(x_\circ, y_\circ, z_\circ)$ is the coordinates of a remote sensor, such as radar and lidar, and r represents the distance between a grid point (x, y, z) and $(x_\circ, y_\circ, z_\circ)$. V_T is the terminal velocity of the precipitation. In a dry atmosphere, V_T is omitted.

The 4DVAR technique converts the constrained minimization problem into an unconstrained problem through the use of the Lagrange function. That is, the constraints including Eqs. (12), (13), the pressure-Poisson equation derived from Eqs. (11) and (12), and the reflectivity conservation equation if applicable, are first multiplied by Lagrange multipliers $\lambda_{\mathcal{F}}$ (also known as adjoint variables), where the subscript $\mathcal{F} = U, V, W, \theta, P$, and Z. They are then appended to the cost function Eq. (14) to form a Lagrange function:

$$\begin{aligned} L = J_o + \sum_t \sum_{x,y,z} [&\lambda_U (x \text{ momentum equation}) + \lambda_V (y \text{ momentum equation}) \\ &+ \lambda_W (z \text{ momentum equation}) + \lambda_\theta (\theta \text{ equation}) + \lambda_P (\text{pressure-Poisson equation}) \\ &+ \lambda_Z (Z \text{ equation})]. \end{aligned} \tag{16}$$

As a result, the constrained minimization of J_o with respect to $\mathcal{F}$ becomes the unconstrained minimization of L with respect to $\mathcal{F}$ and $\lambda_{\mathcal{F}}$. The first variation of L with respect to $\lambda_{\mathcal{F}}$ recovers the governing equations; the first variation of L with respect to $\mathcal{F}$ yields the adjoint equations for the adjoint variables. The integration of these adjoint equations backward in time gives the $\lambda_{\mathcal{F}}$ at initial state. That is,

$$\frac{\partial L}{\partial \mathcal{F}(x, y, z, 0)} = -\lambda_{\mathcal{F}}(x, y, z, 0), \tag{17}$$

where $\mathcal{F} = U, V, W, \theta$, and Z. With these gradients, the limited-memory quasi-Newton algorithm BFGS [4] is applied to find the initial condition for the prediction model for the next iteration. This procedure is repeated until the convergence criterion is satisfied and the resulting solution best fits observational data in a least squares sense. It shall be noted that the above variational approach does not account for model errors. For instance, if the eddy viscosity model is based upon Monin–Obukhov similarity theory, the retrieved data tend to be biased toward satisfaction of the theory that may be inaccurate in some circumstances. Thus, it is desirable to treat the eddy viscosity and diffusivity as control variables as well.

4.3. Smoothness Penalty Constraints

The smoothness constraint P is composed of the temporal smoothness function P^t and the spatial smoothness function P^s: $P = P^t + P^s$. The temporal smoothness function added to the cost function Eq. (14) takes the form

$$P^t = \gamma_U P_U^t + \gamma_V P_V^t + \gamma_W P_W^t + \gamma_\theta P_\theta^t + \gamma_Z P_Z^t, \tag{18}$$

where

$$P^t_{\mathcal{F}} = \sum_{i,j,k} \left(\mathcal{F}^1_{i,j,k} - \mathcal{F}^0_{i,j,k}\right)^2 + \sum_{n\geq 1}\sum_{i,j,k} \left(\mathcal{F}^{n+1}_{i,j,k} - 2\mathcal{F}^n_{i,j,k} + \mathcal{F}^{n-1}_{i,j,k}\right)^2, \tag{19}$$

and the superscript n denotes the time index. The subscripts i, j, and k are the running indices in the respective x, y, and z directions. The spatial smoothness function appended to the cost function reads

$$P^s = \zeta_U P^s_U + \zeta_V P^s_V + \zeta_W P^s_W + \zeta_\theta P^s_\theta + \zeta_Z P^s_Z, \tag{20}$$

where

$$\begin{aligned} P^s_{\mathcal{F}} = \sum_n \sum_{i,j,k} &\left[\left(\mathcal{F}^n_{i+1,j,k} - 2\mathcal{F}^n_{i,j,k} + \mathcal{F}^n_{i-1,j,k}\right)^2 + \left(\mathcal{F}^n_{i,j+1,k} - 2\mathcal{F}^n_{i,j,k} + \mathcal{F}^n_{i,j-1,k}\right)^2\right. \\ &\left. + \left(\mathcal{F}^n_{i,j,k+1} - 2\mathcal{F}^n_{i,j,k} + \mathcal{F}^n_{i,j,k-1}\right)^2\right]. \end{aligned} \tag{21}$$

In the conventional approach, the smoothness penalty coefficients $\gamma_{\mathcal{F}}$ and $\zeta_{\mathcal{F}}$ are determined by trial and error prior to the minimization.

5. FORMULATIONS FOR TUNING SMOOTHNESS PENALTY CONSTRAINTS

In operational applications, the cost function contains the background term and the smoothness penalty constraints.

$$J = J_o + J_b + P, \tag{22}$$

where J_o is defined by Eq. (14). The second term J_b is the background term, measuring the difference between the previous forecast (or analysis) and the data to be retrieved. The reader is referred to Sun and Crook [12] for a description and discussion on this term. The third term P is a smoothness penalty term that consists of the temporal (P^t) and spatial (P^s) smoothness functions.

Let $\mathcal{F}$ denote the model dynamic variables U, V, W, θ, and Z. The formulae for determination of temporal and spatial smoothness penalty coefficients for each dependent variable are given by

$$R^t_{\mathcal{F}} = \frac{\gamma_{\mathcal{F}} P^t_{\mathcal{F}}}{J_o + J_b} \tag{23}$$

and

$$R^s_{\mathcal{F}} = \frac{\zeta_{\mathcal{F}} P^s_{\mathcal{F}}}{J_o + J_b}, \tag{24}$$

where $R^t_{\mathcal{F}}$ designates the ratio of the temporal smoothness penalty constraint over $(J_o + J_b)$ for model variable $\mathcal{F}$. Similarly, $R^s_{\mathcal{F}}$ is for the spatial smoothness constraint. The above two formulae correspond to Step 5 of Algorithm 1, which is to control the level of the smoothness constraints based on the local cost function. Since the smoothness constraints are applied to all dependent variables, one can derive the following relationship:

$$\begin{aligned} \frac{P^t + P^s}{J_o + J_b} &= R^t_U + R^t_V + R^t_W + R^t_\theta + R^t_Z + R^s_U + R^s_V + R^s_W + R^s_\theta + R^s_Z \\ &= R_\circ. \end{aligned} \tag{25}$$

Recall that P^t and P^s are defined by Eqs. (18) and (20). Substitution of Eqs. (18) and (20) into the numerator on the left-hand side of the above equation yields the sum of all ratios $R_\circ$, which is the upper limit of the total smoothness constraint over the total data misfit. $R_\circ$ is normally prescribed as a number less than 1.0 as the previous simple dynamic problem suggests. Each smoothness penalty coefficient can then be determined through Step 7 to Step 10 of Algorithm 1.

The performance of the algorithm for 4DVAR is first evaluated through identical twin experiments (ITE) on the retrieval of micro-scale turbulent structures in an atmospheric convective boundary layer. For the ITE experiments, observational data are generated by the prediction model of the 4DVAR and various sources of error can be added to the observational data to investigate the retrieval sensitivity to the observational error. We then apply the algorithm to retrieve mesoscale atmospheric flow using real Doppler radar data. Evaluation of the algorithm on micro- and mesoscale problems can address the issue about the sensitivity of $R_\circ$ value to the problems of the same nature but different physical scales.

6. APPLICATION TO MICROSCALE FLOW RETRIEVAL USING SYNTHETIC LIDAR DATA

6.1. Generation of Observational Data

The data were first generated by the NCAR-Large-Eddy-Simulation (NCAR-LES) code, which was written by Moeng [6] and improved by Sullivan *et al.* [9] for the study of the atmospheric boundary layer. The results from the NCAR-LES code were used as the initial conditions for the prediction model of the 4DVAR. A computational domain $5 \times 5 \times 2\ \mathrm{km}^3$ is resolved by a grid size $48 \times 48 \times 48$, resulting in spatial resolutions of 104 m and 42 m in the respective horizontal and vertical directions. These are typical lidar range resolutions. The convective boundary layer (CBL) is driven by a geostrophic wind $10\ \mathrm{ms}^{-1}$ and a temperature flux $0.24\ \mathrm{K} \cdot \mathrm{ms}^{-1}$. A capping inversion layer is imposed at $z_i = 980$ m, the CBL height. A Coriolis parameter appropriate to mid latitudes $f = 10^{-4}\ \mathrm{s}^{-1}$ and a roughness height $z_\circ = 0.16$ m are used. The stability parameter $-z_i/L$, where L is the Monin–Obukhov length, is 15. A moving reference frame is applied to both the prediction model and the NCAR-LES code through the Galilean transformation to relax the numerical stability limit and increase the size of time step.

We recorded a total of 13 three-dimensional instantaneous data sets with a time interval of 25 seconds. The vertical distributions of velocity variances of the simulated CBL are exhibited in Fig. 8. Those in a typical CBL measured by Lenschow *et al.* [2] at various stability parameters $-z_i/L$ are also displayed for comparison. They are in good agreement.

The "simulated" velocity fields are then converted to "observed" radial velocity fields through Eq. (15) with $V_T = 0$ because of the dry air. Assume that there is only one lidar located at $(x, y, z) = (0, 0, 21)$ m. These 13 three-dimensional radial data sets are then used to construct two volume scan data sets. Only four horizontal planes of radial velocity data are provided every 25 seconds, sweeping from the surface to the top of the boundary layer twice: from frame 1 to 6 and then from frame 7 to 12. Above the inversion layer ($k > 24$; k, the vertical grid index), turbulence intensity is rather weak, no structures are to be retrieved, and data there are solely provided by frame 13. The assimilation time window is five minutes.

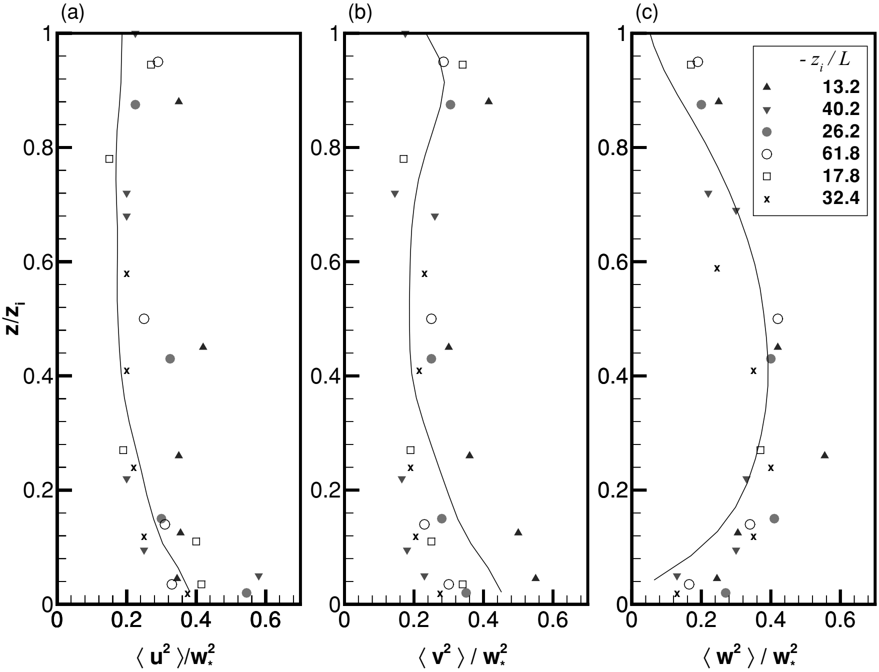

FIG. 8. Vertical distributions of normalized velocity variances for the simulated CBL (solid lines). Field data (denoted by symbols) measured by Lenschow *et al.* (1980) are also displayed for comparison. The free-convection scaling velocity w_* is 2 ms^{-1}.

6.2. Simplification of Formulae

We assume that observational errors are uncorrelated and reflectivity data are not available. Since no data interpolation between different grids is required, the data misfit Eq. (14) can be simplified to the form

$$J_o = \sum_{S,T} \left[\eta_V \left(V_{\text{rad}} - V_{\text{rad}}^o\right)^2\right], \tag{26}$$

where V_{rad}^o denotes the input radial velocity, and V_{rad} is its model counterpart. η_V is the weighting coefficient for radial velocity and is taken as unity. S and T stand for the spatial and temporal extents of the assimilation window. The background term J_b in Eq. (22) is omitted in the ITE experiments.

Lin *et al.* [3] demonstrated that the temporal smoothness penalty constraint becomes less effective in a moving reference frame. Since our ITE experiments are conducted in a reference frame, which moves at a constant speed of 5 ms^{-1} (half the geostrophic wind speed), to extend numerical stability limit and allow a large time step, the temporal smoothness penalty constraint in these experiments is deactivated to reduce the computational cost.

In Algorithm 1 and Eq. (25), the parameter $R_\circ$ is used to control the weighting of smoothness constraints according to the misfit between model data and observational data at each iteration. For the ITE experiments, we impose the following condition:

$$P^s/J_o = R_U^s + R_V^s + R_W^s + R_\theta^s = R_\circ. \tag{27}$$

Note that R_Z^s is omitted because no reflectivity data are available and solving the reflectivity conservation equation is unnecessary. By imposing the smoothness constraint on each dependent variable with a given $R_\circ$, the smoothness penalty coefficients $\zeta_{\mathcal{F}}$ can be determined through Algorithm 1 if the ratios R^s_U, R^s_V, R^s_W, and R^s_θ are specified. For instance, by choosing $R_\circ = 0.5$ and dividing it equally among all variables, we have $R^s_U = R^s_V = R^s_W = R^s_\theta = 0.125$. We first study the cases having the weighting of $R^s_U = R^s_V = R^s_W = R^s_\theta$, then compare them with the cases having different $R^s_{\mathcal{F}}$ values.

The first guess for the ITE experiments is based on the assumption that the horizontal velocity components with a horizontal spatial resolution of 417 m (equivalent to every four grid points) and 20% of random relative errors are measured by a velocity tracking technique [17]. The horizontal velocity at other grid points can be computed using bilinear interpolation. The vertical velocity fluctuation is then derived from the continuity equation [3].

To measure the quality of the retrieved data, the correlation coefficient $\sigma_{\mathcal{F}}$ and the root-mean-square (RMS) error $\epsilon_{\mathcal{F}}$ of the retrieved data at every time step and at every vertical level are calculated. The correlation coefficient is defined as

$$\sigma_{\mathcal{F}} = \frac{\overline{\mathcal{F}' \cdot \mathcal{F}'_\circ}}{\sqrt{\overline{\mathcal{F}'^2}} \cdot \sqrt{\overline{\mathcal{F}'^2_\circ}}}, \tag{28}$$

where $\mathcal{F}'$ designates any retrieved fluctuating velocity component or temperature, and $\mathcal{F}'_\circ$ represents its exact counterpart. The overline denotes spatial averaging over an $x - y$ plane.

6.3. Results

The results from the ITE experiments after 30 iterations are tabulated in Table I. The correlation coefficient between retrieved data and exact data and the RMS error of retrieved data in Table I are averaged throughout the boundary layer. In these experiments, we degrade the quality of observational data by adding random errors to the radial velocity. Four different error amplitudes, $A_\epsilon = 0.0$, 0.5, 1.0, and 1.5 ms^{-1}, are used. The random error with $A_\epsilon = 0.5$ ms^{-1} (an RMS value of 0.29 ms^{-1}) roughly corresponds to the error present in a measurement taken on a day with the very clear air and averaging on 100 lidar pulses over a range of 3 km.

For cases ITE01, ITE02, and ITE03 without observational errors, we see that $R_\circ = 0.5$ yields better results than that without penalizing smoothness constraints, while $R_\circ = 1.0$ has negligible effect on retrieval. With $A_\epsilon = 0.5$ ms^{-1}, case ITE05 with $R_\circ = 0.2$ retrieves data of about the same quality as case ITE08, which uses the fixed smoothness penalty coefficients: $\zeta_U = \zeta_V = \zeta_W = 0.00005$ and $\zeta_\theta = 0.001$. Note that these fixed coefficients are the optimal values obtained by trial and error [3]. Case ITE06 with $R_\circ = 0.5$ yields slightly less accurate results than case ITE05 with $R_\circ = 0.2$, while case ITE07 suggests that the ratio $R_\circ = 1.0$ produces an over-smoothness effect. With increasing error amplitude $A_\epsilon = 1.0$ and 1.5 ms^{-1}, the results show consistently that the ratio $R_\circ = 0.2$ yields better results than other $R_\circ$ values and outperforms cases using the fixed penalty coefficients.

Figure 9 shows the variation of the smoothness penalty coefficient with respect to the number of iterations for case ITE10, which uses $R_\circ = 0.2$ and $A_\epsilon = 1.0$ ms^{-1}. A comparison with the fixed coefficients (Fig. 9) finds that the algorithm generates larger smoothness coefficients at the early stage of assimilation, possibly acting to accelerate convergence by reducing irregular curvatures as illustrated by the solid curves versus the dashed curves in

TABLE I
Identical Twin Experiments

Case	A_ϵ	$R_\circ{}^a$	σ_U	σ_V	σ_W	σ_θ	ϵ_U	ϵ_V	ϵ_W	ϵ_θ
ITE01	0.0	0.0	0.93	0.94	0.89	0.69	0.33	0.36	0.42	0.20
ITE02	0.0	0.5	0.94	0.95	0.93	0.71	0.30	0.32	0.35	0.19
ITE03	0.0	1.0	0.92	0.94	0.90	0.68	0.36	0.38	0.45	0.18
ITE04	0.5	0.0	0.91	0.91	0.86	0.67	0.37	0.40	0.49	0.21
ITE05	0.5	0.2	0.92	0.93	0.90	0.69	0.34	0.37	0.41	0.20
ITE06	0.5	0.5	0.91	0.93	0.87	0.67	0.37	0.40	0.49	0.18
ITE07	0.5	1.0	0.86	0.88	0.80	0.63	0.49	0.51	0.62	0.18
ITE08	0.5	fixed[b]	0.92	0.93	0.89	0.70	0.35	0.37	0.42	0.19
ITE09	1.0	0.0	0.85	0.87	0.78	0.64	0.48	0.51	0.64	0.23
ITE10	1.0	0.2	0.89	0.90	0.86	0.66	0.42	0.44	0.50	0.19
ITE11	1.0	0.5	0.86	0.89	0.81	0.63	0.48	0.50	0.61	0.18
ITE12	1.0	fixed[b]	0.87	0.89	0.83	0.66	0.45	0.47	0.55	0.21
ITE13	1.5	0.0	0.80	0.82	0.72	0.59	0.56	0.59	0.75	0.24
ITE14	1.5	0.2	0.86	0.88	0.82	0.64	0.47	0.49	0.57	0.19
ITE15	1.5	0.5	0.84	0.87	0.77	0.62	0.51	0.53	0.65	0.18
ITE16	1.5	fixed[b]	0.82	0.83	0.75	0.61	0.54	0.57	0.68	0.22

A_ϵ, random error amplitude ms^{-1}; $\sigma_\mathcal{F}$, correlation coefficient between retrieved data and exact data; $\epsilon_\mathcal{F}$, RMS error of retrieved data in ms^{-1} for velocity and in K for temperature. $\sigma_\mathcal{F}$ and $\epsilon_\mathcal{F}$ are averaged throughout the boundary layer.

[a] All the cases except those with "fixed[b]" assume $R_U^s = R_V^s = R_W^s = R_\theta^s$.

[b] Fixed smoothness coefficients: $\zeta_U = \zeta_V = \zeta_W = 0.00005$ and $\zeta_\theta = 0.001$.

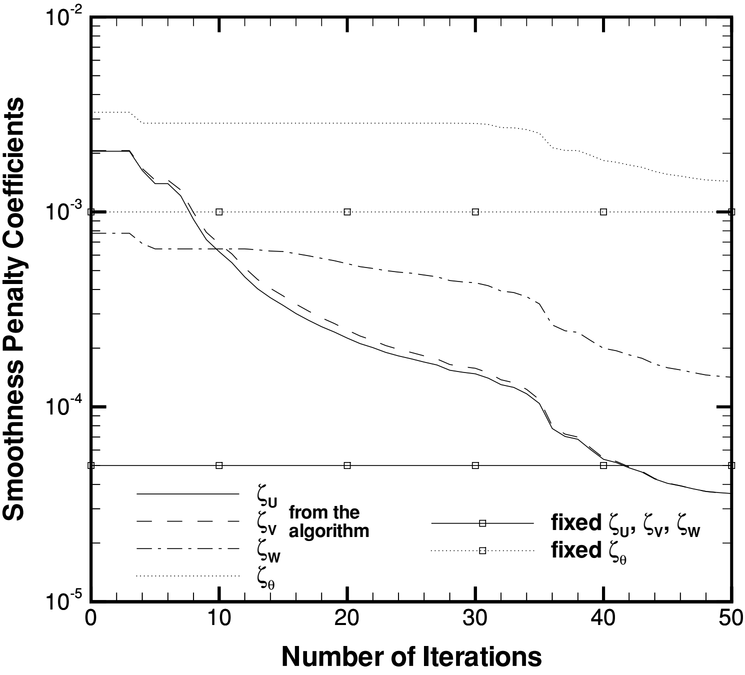

FIG. 9. Variations of the smoothness penalty coefficients $\zeta_\mathcal{F}$ with respect to the number of iterations for case ITE10. The fixed $\zeta_\mathcal{F}$ values used in case ITE12 are also shown for comparison.

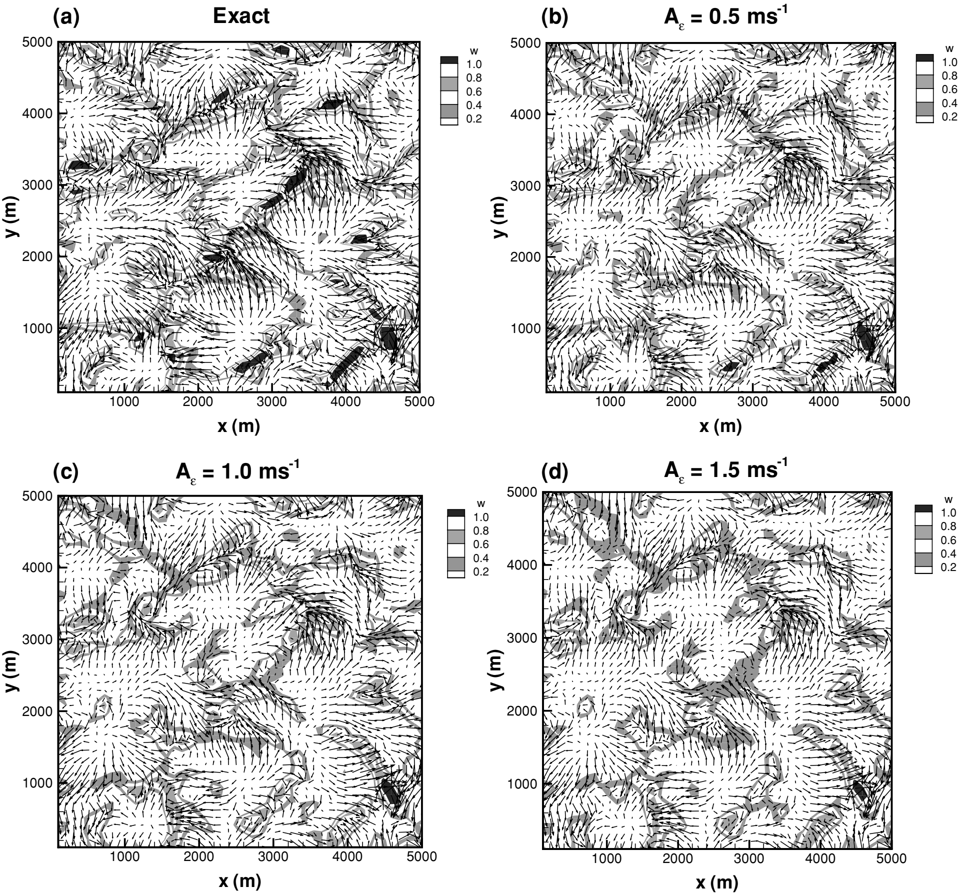

FIG. 10. Horizontal fluctuating velocity vectors and contours of vertical fluctuating velocity in ms^{-1} at $z/z_i = 0.085$ for cases (b) ITE05, (c) ITE10, (d) ITE14, which use $R_\circ = 0.2$. The corresponding exact data are shown in (a). A_ϵ, random error amplitude.

Fig. 2b. As the solution improves with increasing iterations, the coefficients reduce to about the same order of magnitudes as the fixed coefficients (Fig. 9) and keep decreasing with iterations. This is an appealing feature because decreasing coefficients indicate the smoothness constraint being maintained at a desired level with respect to the local cost function and shall lead to more accurate solutions. Figure 10 displays the horizontal fluctuating velocity vectors and the contours of the vertical fluctuating velocity near the surface for cases ITE05, ITE10, and ITE14. These cases use the same $R_\circ = 0.2$ and contain random errors of various amplitude. We see that most of flow structures, such as convergence lines and divergence regions, are retrieved regardless of the error amplitude.

In what follows we investigate the effect of differing $R^s_{\mathcal{F}}$. The ratios $R^s_U : R^s_V : R^s_W : R^s_\theta$ for cases ITV01, ITV02, ITV04, and ITV05 in Table II are roughly weighted by the availability of data. It is based on the principle that if there are more observational data, the minimization relies less on the smoothness penalty constraint. Since there is no temperature data, a stricter smoothness constraint is imposed on temperature. In the current CBL, which is driven by a geostrophic wind of 10 ms^{-1} and a temperature flux, the U velocity component is expected to be larger than the V and W velocity components and the radial velocity possibly contains

TABLE II
Identical Twin Experiments

Case	$R_\circ$	$R_U^s : R_V^s : R_W^s : R_\theta^s$	σ_U	σ_V	σ_W	σ_θ	ϵ_U	ϵ_V	ϵ_W	ϵ_θ
ITV01	0.2	1 : 1 : 1 : 2	0.85	0.87	0.82	0.64	0.49	0.51	0.56	0.19
ITV02	0.2	2 : 3 : 3 : 12	0.84	0.86	0.81	0.65	0.50	0.52	0.56	0.19
ITV03	0.2	$P_U^s : P_V^s : P_W^s : P_\theta^s$	0.86	0.88	0.80	0.62	0.47	0.50	0.60	0.19
ITV04	0.5	1 : 1 : 1 : 2	0.84	0.87	0.78	0.62	0.50	0.52	0.63	0.18
ITV05	0.5	2 : 3 : 3 : 12	0.85	0.88	0.80	0.65	0.48	0.50	0.61	0.17
ITV06	0.5	$P_U^s : P_V^s : P_W^s : P_\theta^s$	0.84	0.87	0.77	0.61	0.51	0.53	0.66	0.18

The $R_{\mathcal{F}}^s$ values depend on the $R_\circ$ value and the ratio $R_U^s : R_V^s : R_W^s : R_\theta^s$. $A_\epsilon = 1.5$ ms^{-1}.

more information about U than V and W. Thus, cases ITV02 and ITV05 place more weights on R_V^s and R_W^s than R_U^s. We first apply the above principle to temperature in cases ITV01 and ITV04, then extend it to velocity as well in cases ITV02 and ITV05. The above guideline merely suggests the relationship of $R_U^s \leq R_V^s \approx R_W^s \leq R_\theta^s$. The specification of their values remains arbitrary. It is found that case ITV05 can yield better results than ITE15 in Table I, which uses the same $R_\circ$ value but imposes uniformly the smoothness constraints on different variables. Nonetheless, the same weighting does not apply to different $R_\circ$ values, e.g., case ITV02 versus case ITE14 in Table I. Cases ITV01 and ITV04 do not produce better results than cases ITE14 and ITE15 either. As for case ITV03 and ITV06, a different weighting scheme is examined. The $R_{\mathcal{F}}^s$ values in these cases are weighted by the degree of smoothness of each variable, measured by the penalty function $P_{\mathcal{F}}^s$, Eq. (21). It is based on the hypothesis that a larger $P_{\mathcal{F}}^s$ (less smooth $\mathcal{F}$) requires a stricter smoothness penalty constraint. This can be achieved by assuming that the values of $\zeta_{\mathcal{F}}$ for all variables are the same and are equal to c. The c value at each iteration is obtained through $c(P_U^s + P_V^s + P_W^s + P_\theta^s)/J_\circ = R_\circ$. The results of the two cases (Table II) are not as good as the uniform constraint cases.

In general, we can draw the following conclusions. First, by elaborate tuning of the $R_{\mathcal{F}}^s$ values, it is possible to get better results. However the optimal weighting of $R_{\mathcal{F}}^s$ for one $R_\circ$ does not necessarily apply to a different $R_\circ$. Second, the use of uniform smoothness constraints on different variables yields consistently good results regardless of the $R_\circ$ value. Therefore, in the next section regarding the applications of the method to meso-scale flow retrieval, we will apply the smoothness constraint uniformly on different variables.

Before closing this section, we shall discuss the role played by the condition $R > R_\circ$ at step 7 in Algorithm 1. Consider the variation of the penalty coefficient $\zeta_{\mathcal{F}}$ for case ITE10 shown in Fig. 9. In the first few iterations, the condition $R > R_\circ$ is not satisfied and the coefficients $\zeta_{\mathcal{F}}$ are not updated and remain constant. Without this condition, a fixed ratio $R_\circ$ is strictly enforced at each iteration and $\zeta_{\mathcal{F}}$ are updated accordingly. As a result, all of the $\zeta_{\mathcal{F}}$ values increase in the first few iterations and the minimization process is terminated abruptly. The problem, of course, is attributable to the violation of the global convergence criterion discussed in Section 3. It is also observed that an over-smoothed first guess for the initial conditions results in small values of the penalty functions $P_{\mathcal{F}}^s$, which then generate large $\zeta_{\mathcal{F}}$. The large $\zeta_{\mathcal{F}}$ subsequently produce much smoother $\mathcal{F}$ model data. This adverse cycle continues to produce larger and larger penalty coefficients as the iteration goes on.

In the quasi-Newton L-BFGS method, two conditions are used to determine a step length in a given search direction: a sufficient decrease condition and a curvature condition [7]. The first guess for the initial conditions of model variables U, V, W, θ usually is estimated from the mean or relatively large-scale observations. The over-smoothness of the first guess, the increasing $\zeta_{\mathcal{F}}$ described above, and the satisfaction of the above step-length conditions can result in a large step length. As a consequence, the over corrections to the initial conditions occur and the numerical instability follows. Thus, the condition $R > R_\circ$ at step 7 in the algorithm is to avoid over-smoothness on variables that have sparse or no observational data, which tend to cause the above problem. In spite of the imposed condition $R > R_\circ$, Fig. 9 shows that the preferred condition $R = R_\circ$ is satisfied in most of iterations. We also tried cases that impose the condition $R > R_\circ$ only in the first 10 iterations. But the minimization also stopped later, suggesting that the condition $R > R_\circ$ should be enforced throughout the minimization to ensure stability.

7. APPLICATION TO MESOSCALE FLOW RETRIEVAL USING REAL DOPPLER RADAR DATA

In order to examine the performance of the algorithm using real data, we apply the method to a four-dimensional variational Doppler radar analysis system (VDRAS). VDRAS was designed to assimilate a time series of radar observations (radial velocity and reflectivity) from single or multiple Doppler radars. For a detailed presentation of the system and its real-time application, the reader is referred to Sun and Crook [14]. Here we only provide a brief description. The constraining numerical model is similar to that described in Section 4.1. There are five prognostic equations: one for each of the three velocity components U, V, W, the potential temperature fluctuation θ, and the reflectivity Z. The pressure is diagnosed through a Poisson equation. By fitting the model to observations over a specified time period, a set of optimal initial conditions for the constraining numerical model can be obtained.

The cost function, which measures the misfit between the model variables and both the observations and a prior estimate, is defined by Eq. (22). The first term of Eq. (22) J_o represents the discrepancy from the radar observations. Since we neglect errors in the observations and in the observation operator, the function J_o [Eq. (14)] is expressed as

$$J_o = \sum_{S,\,T} \left\{ \eta_V \left[H(V_{\text{rad}}) - V^o_{\text{rad}} \right]^2 + \eta_Z [H(Z) - Z^o]^2 \right\}. \tag{29}$$

As compared with Eq. (26) for ITE, an additional term appears on the right-hand side of the above equation due to the availability of reflectivity data Z. The superscript o denotes the observations, and V_{rad} is related to the Cartesian velocity components through Eq. (15). Since the model data is displayed on a Cartesian grid and the radar data is on a spherical grid, the model radial velocity and the reflectivity must be transformed to the spherical grid by an observation operation H [14]. The diagonal entries of the matrix $\mathbf{O}^{-1}$ in Eq. (14) η_V and η_Z are the weighting coefficients for radial velocity and reflectivity, respectively, and are assigned unity. Reflectivity is one of the model prognostic variables while the model radial velocity has to be computed from the model Cartesian velocity components through Eq. (15), in which the terminal velocity V_T is estimated using the reflectivity data.

The background term J_b in Eq. (22) measuring the discrepancy from the previous analysis or forecast is incorporated in the real data experiments. The reader is referred to Sun and Crook [12–14] for a description and discussion on this term. The formulae for tuning the temporal and spatial smoothness penalty coefficients are given by Eqs. (23) and (24).

Since the previous ITE experiments show that $R_\circ = 1.0$ tends to over-smooth the data, the real data experiments are conducted using different values of $R_\circ$ in the range of 0.1 to

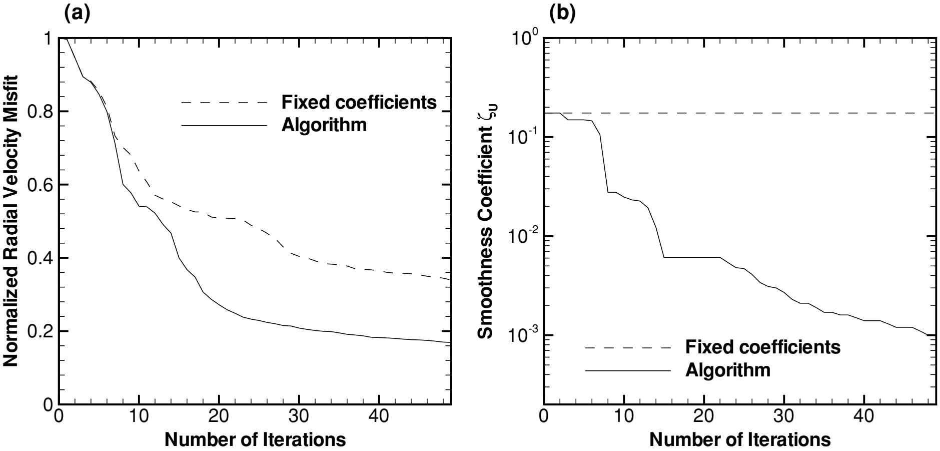

FIG. 11. (a) Reduction of the radial velocity misfit with respect to number of iterations. (b) Variation of the smoothness coefficient ζ_U with number of iterations.

0.5. We evenly distribute the value of $R_\circ$ to the ratios of different dynamic model variables: $R^t_{\mathcal{F}} = R^s_{\mathcal{F}} = R_\circ/10$ where $\mathcal{F} = U, V, W, \theta$, and Z.

The radar data used in the experiments were observed by the WSR-88D KLWX radar located at Sterling, Virginia, in the afternoon of June 15, 1998. The data depict a strong thunderstorm outflow that propagates southeastward at a speed of around 13 ms^{-1}. In the numerical model, we use a grid resolution of 3 km in the horizontal and 375 m in the vertical with a grid mesh of 50 × 50 × 7. The assimilation time window is 10 min, which covers three radar volume scans. By iteratively minimizing the cost function Eq. (22) using the optimization procedure described in Section 4.2, an optimal set of the model variables can be obtained. The optimal solution matches the observations as closely as possible and satisfies the constraining equations.

We evaluate the quality of the retrieval using the radial velocity misfit and the subsequent forecast initialized by the retrieved fields. The forecast is verified by the radial velocity observations. All the experiments with different values of $R_\circ$ in the range of 0.1 to 0.5 show an improved fit to the radial velocity observations as compared with the fixed coefficient case. Figure 11a shows the reduction of the radial velocity misfit with respect to the number of iterations from the experiments with $R_\circ = 0.3$. The solid curve is from the experiment with the algorithm and the dashed curve with fixed penalty coefficients. The penalty coefficients ζ_U from these two experiments are shown in Fig. 11b. As observed from this figure, the coefficient determined by the algorithm reduces rapidly in a few iterations, resembling those in the microscale retrieval (Fig. 9). As the penalty coefficients continue to decrease in magnitude with iterations, the minimization is able to find a closer fit to the observations as indicated by the solid line in Fig. 11a. This is not surprising by considering the principle of the modified Newton method, which is to reduce the influence of the matrix $\mathbf{E}$ in Eq. (9) near the minimizer to obtain a more accurate solution.

To examine whether the retrieval with a closer fit to the radial velocity observations improves the subsequent forecast, two forecasts were produced using the retrieved fields from the experiments with fixed penalty coefficients and with the algorithm, respectively. To verify the forecast, the radial velocity from the forecast was interpolated to the grid of the radial velocity observations, and the RMS error was calculated. The boundary conditions at

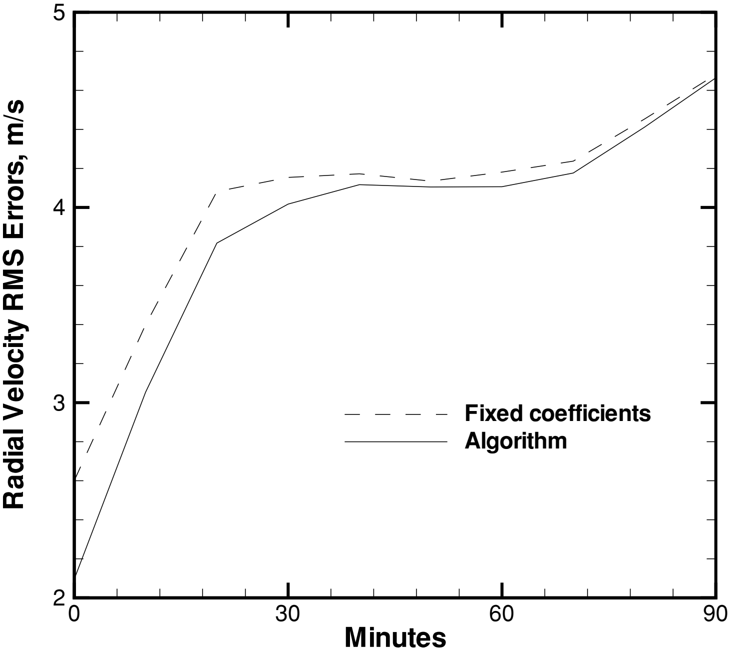

FIG. 12. Time history of radial velocity RMS error.

the initial time were applied throughout the entire forecast period. The radial velocity RMS errors from these two forecasts are shown in Fig. 12. The results show that the forecast utilizing the proposed algorithm has a smaller RMS error throughout the forecast period. It should be noted that since the domain of the retrieval and the forecast is only 150 km, the error in the boundary conditions can have a rather large influence on the forecast. As a result, the forecast error converges to the same value as the forecast time increases.

Similar forecast experiments were also conducted with other values of $R_\circ$. The results show that when $R_\circ$ equals 0.1 or 0.2, the forecast with the algorithm produced larger RMS forecast error than that with the fixed coefficients although the retrieval fits to the observations much closer. It indicates that the long-time evolution of smaller scale features in the retrieved fields obtained by fitting closer to the observations may not be well predicted by the numerical model, hence they become noise that degrades the forecast. Our experiments show that $R_\circ = 0.3$ produces the best forecast. In comparison with the value $R_\circ = 0.2$ found in Section 6, it seems to suggest that a slightly larger value of $R_\circ$ should be used with real data for the purpose of forecast.

It is quite encouraging to find that the optimal $R_\circ$ values for both microscale and mesoscale applications are about the same. In contrast, the fixed coefficient approach uses quite different values of penalty coefficients for microscale and mesoscale retrieval, e.g., 0.00005 [3] and 0.05 [11], respectively. The algorithm is able to control the degree of smoothness constraint throughout minimization and generates more accurate results, attributable to the satisfaction of the global convergence criterion, Eq. (10). A potential application of the algorithm is in the development of an adaptive 4DVAR technique that integrates data of various sources at different scales, such as lidar and radar data. While passing and assimilating data through different grid levels, the smoothness constraints can be applied in a more consistent and uniform manner without elaborate tuning of the weights of the smoothness constraints.

8. CONCLUSIONS

The use of smoothness penalty constraints in the assimilation of atmospheric and oceanic data into dynamic models is known to improve the conditioning of the minimization problems. The coefficients of the smoothness constraints, however, could be several orders of magnitude different for different problems. To devise an effective way to determine the weights, we have studied a simple dynamic system. It is found that penalizing smoothness constraints makes the modified Hessian matrix of the cost function more positive definite. The concept is akin to the modified Newton methods that modify the Hessian matrix of the objective function to make it more positive definite and better conditioned. Due to the nature of the smoothness penalty function, there is no warranty that the constraints can be uniformly applied at every iteration during the minimization process. One strategy to control the level of smoothness locally is to adjust the weights of smoothness according to the local cost function by fixing the ratio of the smoothness constraints over the cost function. Although the approach of fixed ratio makes the modified Hessian matrix much more positive definite, the condition required for the global convergence may not be satisfied due to random large weights resulted. Based on these observations, we propose an algorithm that is able to determine the weights of smoothness constraints following the idea of fixed ratio but ensures monotonic decrease of the smoothness coefficients.

We first apply the algorithm to a simple harmonic oscillator problem, then test it on the retrieval of microscale turbulent structures in the atmospheric convective boundary layer through the approach of identical twin experiments. The identical twin experiments allow assessment of the sensitivity of the proposed method to the amplitude of observational error. The results indicate that a ratio $R_\circ$ around 0.2 tends to give better results regardless of the error amplitude.

We finally apply this method to the Doppler radar data, depicting a strong thunderstorm outflow. The results show consistently that the algorithm yields more accurate retrieval than the approach of fixed coefficients. To examine whether the retrieval with a closer fit to the observations improves the subsequent forecast, we use the retrieved data as an initial condition for the forecast model to predict the atmospheric state. The results suggest that a slightly larger value of $R_\circ = 0.3$ should be used to get a better forecast. A larger $R_\circ$ imposes stronger smoothness constraints on small-scale flow structures, suggesting that the presence of small-scale structures is not favorable to prediction. Since the optimal $R_\circ$ values for both microscale and mesoscale applications are about the same, the algorithm may effectively reduce the effort in tuning the smoothness weights when applying the same 4DVAR system with a single value of $R_\circ$ to the problems of the same nature but varying physical scales.

In summary, we have applied the proposed algorithm first to a simple harmonic oscillator problem, with increasing complexity, to the retrieval of microscale turbulent structures using synthetic lidar data, and then to the retrieval of a mesoscale severe thunderstorm outflow using real Doppler radar data. All the experiments indicate that the proposed algorithm, which tunes the smoothness penalty constraints based on data misfit at every iteration, produces better results than the conventional approach using constant smoothness coefficients.

ACKNOWLEDGMENTS

This work is supported by a National Science Foundation ATM Grant 9874925, monitored by Dr. Roddy R. Rogers. Acknowledgment is made to the National Center for Atmospheric Research, the National Partnership for Advanced Computational Infrastructure, and the National Center for Supercomputing Applications, which are sponsored by the National Science Foundation, for the computing time used in the research.

REFERENCES

1. R. N. Hoffman, A four-dimensional analysis exactly satisfying the equations of motion, *Mon. Weather Rev.* **104**, 1551 (1985).
2. D. H. Lenschow, J. C. Wyngaard, and W. T. Pennell, Mean-field and second-moment budgets in a baroclinic, convective boundary layer, *J. Atmos. Sci.* **37**, 1313 (1980).
3. C.-L. Lin, T. Chai, and J. Sun, Retrieval of flow structures in a convective boundary layer using an adjoint model: Identical twin experiments, *J. Atmos. Sci.* **58**, 1767 (2001).
4. D. C. Liu and J. Nocedal, On the limited memory BFGS method for large-scale optimization, *Math. Program* **45**, 503 (1989).
5. R. B. Long and W. C. Thacker, Data assimilation into a numerical equatorial ocean model. 2. Assimilation experiments, *Dyn. Atmos. Oceans* **13**, 413 (1989).
6. C.-H. Moeng, A large-eddy-simulation model for the study of planetary boundary-layer turbulence, *J. Atmos. Sci.* **41**, 2052 (1984).
7. J. Nocedal and S. J. Wright, *Numerical Optimization* (Springer-Verlag, Berlin/New York, 2000).
8. K. V. Ooyama, Scale-controlled objective analysis, *Mon. Weather Rev.* **115**, 2479 (1987).
9. P. P. Sullivan, J. C. McWilliams, and C.-H. Moeng, A subgrid-scale model for large-eddy simulation of planetary boundary-layer flows, *Boundary-Layer Meteorol.* **71**, 247 (1994).
10. J. Sun, D. W. Flicker, and D. K. Lilly, Recovery of three-dimensional wind and temperature fields from simulated single-Doppler radar data, *J. Atmos. Sci.* **48**, 876 (1991).
11. J. Sun and A. Crook, Wind and thermodynamic retrieval from single-Doppler measurements of a gust front observed during Phoenix II. *Mon. Weather Rev.* **122**, 1075 (1994).
12. J. Sun and A. Crook, Dynamical and microphysical retrieval from Doppler radar observations using a cloud model and its adjoint. I: Model development and simulated data experiments, *J. Atmos. Sci.* **54**, 1642 (1997).
13. J. Sun and A. Crook, Dynamical and microphysical retrieval from Doppler radar observations using a cloud model and its adjoint. II. Retrieval experiments of an observed Florida Convective Storm, *J. Atmos. Sci.* **55**, 835 (1998).
14. J. Sun and N. A. Crook, Real-time low-level wind and temperature analysis using WSR88D data, Submitted to Weather and Forecasting.
15. Thacker, handwriting class notes, unpublished.
16. W. C. Thacker, Fitting models to inadequate data by enforcing spatial and temporal smoothness, *J. Geophys. Res.* **93**, 10655 (1988).
17. J. D. Tuttle and G. B. Foote, Determination of the boundary layer airflow from a single Doppler radar, *J. Atmos. Oceanic Tech.* **7**, 218 (1990).
18. B. Wu, J. Verlinde, and J. Sun, Dynamical and microphysical retrievals from Doppler radar observations of a deep convective cloud, *J. Atmos. Sci.* **57**, 262 (2000).
19. S. Yang and Q. Xu, Statistical errors in variational data assimilation—A theoretical one-dimensional analysis applied to Doppler wind retrieval, *J. Atmos. Sci.* **53**, 2563 (1996).