

Application of the Multigrid Method and a Flexible Hybrid Coordinate in a Nonhydrostatic Model

SHU-HUA CHEN

National Center for Atmospheric Research, Boulder, Colorado

WEN-YIH SUN

Department of Earth and Atmospheric Sciences, Purdue University, West Lafayette, Indiana

(Manuscript received 7 September 2000, in final form 4 June 2001)

ABSTRACT

A fully compressible, three-dimensional, nonhydrostatic model is developed using a semi-implicit scheme to avoid an extremely small time step. As a result of applying the implicit scheme to high-frequency waves, an elliptic partial differential equation (EPDE) has been introduced. A multigrid solver is applied to solve the EPDEs, which include cross-derivative terms due to terrain-following coordinate transformation.

Several experiments have been performed to evaluate the model as well as the performance of the scheme with respect to tolerance number, relaxation choice, sweeps of prerelaxation and postrelaxation, and a flexible hybrid coordinate (FHC).

An FHC with two functions (base and deviation functions) is introduced. The basic function provides constant vertical grid spacing required in the multigrid solver, while the deviation function helps to adjust the vertical resolution.

1. Introduction

In the past, numerical weather prediction models had to use the governing equations with the hydrostatic approximation in order to avoid acoustic waves in the vertical direction. When high-frequency acoustic waves exist, an exceedingly small time step is required to prevent numerical instability of the explicit time-stepping scheme (Mendez-Nunez and Carroll 1994; Hsu and Sun 2001). Fully compressible nonhydrostatic models, which include sound waves, have become more popular in recent atmospheric modeling due to the improvement of numerical techniques and the rapid increase of computing resources. The explicit time-splitting scheme (Gadd 1978) implemented in nonhydrostatic models (Klemp and Wilhelmson 1978) has significantly reduced computational time. The development of the semi-implicit scheme (Robert 1969) was a remarkable milestone in the development of numerical techniques. The use of the semi-implicit scheme in nonhydrostatic models has relaxed the extremely small time step constraint (Tapp and White 1976; Tanguay et al. 1990; Cullen 1990; Juang 1992; Golding 1992; Kapitza and Eppel 1992;

Robert 1993; Pinty et al. 1995; Simmons and Temperton 1996; Skamarock et al. 1997).

In some fully compressible nonhydrostatic models, only acoustic waves are calculated implicitly (Tapp and White 1976; Juang 1992; Laprise 1992), and the time step is constrained by the propagation speeds of gravity waves. Here, we apply the implicit scheme to both acoustic and gravity waves, as in Tanguay et al. (1990), Cullen (1990), Golding (1992), and Pinty et al. (1995). It is noted that our model has differences from theirs because we keep the cross-derivative terms, due to the terrain-following coordinate, in the elliptic partial differential equation (EPDE) of the Exner function perturbation. Usually, a huge multiband sparse matrix ($\sim 300\,000 \times 300\,000$) of the EPDEs is encountered for high-resolution nonhydrostatic models. Scientists have applied different methods to avoid calculating this huge multiband sparse matrix. With the use of the explicit time-splitting scheme, only vertical acoustic waves are calculated implicitly, while horizontal acoustic waves (Lamb waves) are calculated explicitly. Under this condition, the horizontal resolution should not be too high in order to avoid the requirement of a small time step. Some scientists (Juang 1992; Cullen 1990; Golding 1992; Pinty et al. 1995) also suggest integrating the cross-derivative terms explicitly in order to simplify the equations to become Poisson- or Helmholtz-type equations, which are more easily solved. However, nu-

Corresponding author address: Dr. Shu-hua Chen, National Center for Atmospheric Research, P.O. Box 3000, Boulder, CO 80307.
E-mail: shchen@ucar.edu

merical instability may occur over steep mountains with this kind of approach (Ikawa 1988; Skamarock et al. 1997).

There are different ways to solve linear EPDEs mathematically, including direct methods [Gaussian elimination, Fourier and cyclic reduction, and lower-upper (LU) triangular decomposition] and iterative methods [Jacobi, Gauss–Seidel, successive overrelaxation (SOR), alternative direction implicit (ADI), conjugate gradient, and multigrid]. The advantages and disadvantages of different methods have been briefly discussed in Fulton et al. (1986) and Kapitza and Eppel (1992). The ADI scheme works well if there is no cross-derivative term (Cullen 1990; Golding 1992; Pinty et al. 1995). To integrate cross-derivative terms implicitly, Kapitza and Eppel (1992) used the conjugate gradient method, which is usually applied to symmetric problems, with a preconditioner, and Skamarock et al. (1997) used the conjugate residual method with a preconditioner to solve the EPDEs. The convergence rate of the EPDEs depends upon the condition number of iteration matrix, which is the largest eigenvalue divided by the smallest one, and the number of distinct eigenvalues. A good preconditioner can cluster the eigenvalues and result in a faster convergence rate. Considerable experimentation is necessary to find a good preconditioner.

The multigrid method is a very general and efficient method for elliptic-type or Helmholtz-type problems (Brandt 1977; Fulton et al. 1986; Ciesielski et al. 1986; Bates et al. 1990; Bowman and Huang 1991; Adams et al. 1992; Engquist and Luo 1997; Mertens et al. 1998; Hiptmair 1998; Adams and Smolarkiewicz 2001). This method is a nested iteration scheme that was originally applied to boundary value problems. It is well known that the convergence rate of standard iterative methods, such as Jacobi, Gauss–Seidel, and SOR, are related to the eigenvalues of the iterative matrix. For linear discretized partial differential equations, the eigenvalues (convergence rates) are a function of the grid resolution. These standard iterative methods rapidly eliminate the error associated with short waves, but leave the long waves almost unchanged. The iteration on a coarse grid is much cheaper and more efficient since fewer points are involved. This feature becomes the essence of the multigrid method, which uses a coarse grid correction (long-wave correction) to improve the convergence rate. Different orders of relaxation cycling schemes can be used, as shown in Fig. 1. A V cycle (Fig. 1a) or W cycle (Fig. 1b) is preferred if a good initial guess is provided. A full multigrid V cycle (Fig. 1c), which starts from the coarsest grids, is suggested if there is no good initial guess. Four processes are involved in this method: prerelaxation, coarse grid solution, coarse grid correction, and postrelaxation. Figure 2 shows the procedure of a three-level $V(N_1, N_2)$ -cycle multigrid method, where N_1 and N_2 are the numbers of relaxation sweeps for

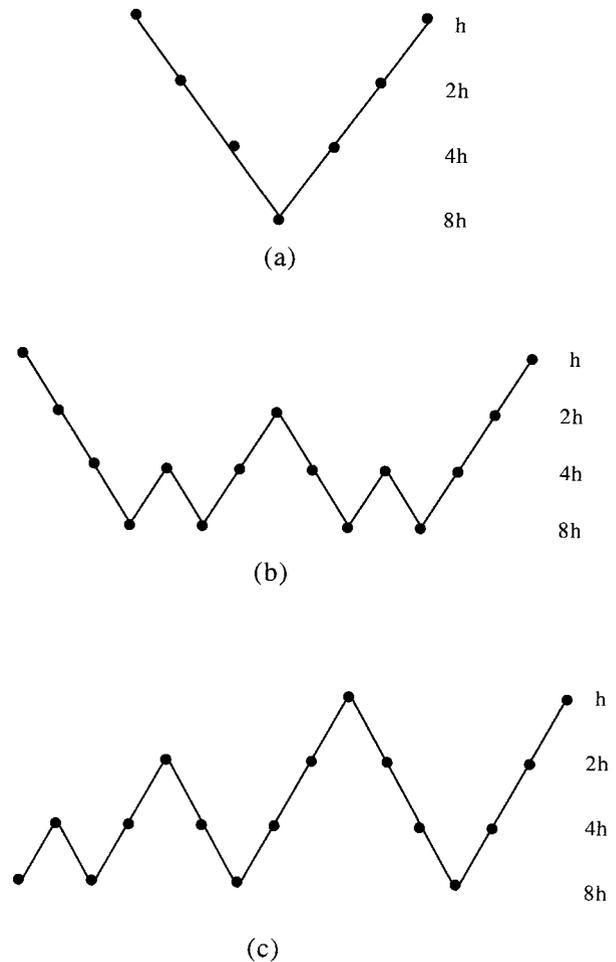


FIG. 1. Schedule of grids (from left to right) for (a) V cycle, (b) W cycle, and (c) full multigrid V cycle schemes on four levels (adopted from Fig. 19 of Briggs 1987). Here, h is the finest grid level and $8h$ is the coarsest grid level.

prerelaxation and postrelaxation, respectively, and a brief description is given in appendix A.

As mentioned earlier, the implicit scheme is applied to both acoustic and gravity waves, and therefore a set of linear equations with a huge multiband sparse matrix needs to be solved. Modelers have been looking for a method, which is able to solve the EPDEs accurately and efficiently, and which can be applied to calculate this linear system with less constraint, such as consistently integrating the cross-derivative terms implicitly. Though the conjugate gradient method was used, Kapitza and Eppel (1992) mentioned that the iterative multigrid technique is the most efficient algorithm for Poisson-type problems. However, they argued that the multigrid method suffers from the requirement that the number of grid intervals must be a power of 2. In fact, this constraint can be slightly relaxed if the matrix can be reasonably calculated by direct solvers or by repeated relaxation sweeps on the coarsest grids. It seems that the multigrid method has high potential for effi-

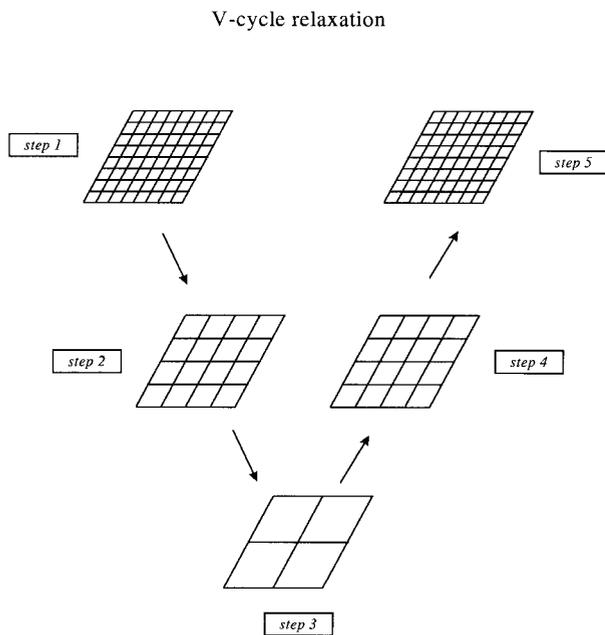


FIG. 2. The procedure of a three-level $V(N_1, N_2)$ cycle multigrid method.

ciently solving the linear EPDEs. This method is not as popular in the atmospheric sciences as are the conjugate gradient and conjugate residual methods. More investigation might be both interesting and necessary in order to learn the characteristics of this method and to learn the appropriateness of applying it to atmospheric models. For these reasons, the multigrid method is chosen for use in this study, with the primary objective being to investigate the efficiency and accuracy of this method.

The multigrid solver developed by Adams [mud3cr; Adams and Smolarkiewicz (2001)] is applied to this study since this package is quite comprehensive. (This solver is available on the World Wide Web at <http://www.scd.ucar.edu/css/software/mudpack>.) Mud3cr computes the second-order finite-difference approximation to three-dimensional linear nonseparable EPDEs with cross-derivative terms. A Gauss–Seidel iteration is applied to each level and a block Gaussian elimination method is used to solve the linear equations at the coarsest grid level. The full weighting restriction and the multilinear interpolation (appendix A) are used. The size of grid intervals is not restricted to a power of 2 (Adams 1989), but still needs to satisfy the formula of $2^x \times y$, where x is a non-negative integer and y is a prime number. Small prime numbers are recommended as described in the package. A four-color ordering scheme, which makes the solver converge faster and makes it suitable for parallelizing, is implemented. The convergence criteria are defined as a function of the maximum vector norm, and are introduced in section 3. Many options are provided in this package, and several fea-

tures are tested in this study, such as different relaxation choices (point by point or line by line), different tolerance numbers, and different sweeps of prerelaxation and postrelaxation. The choice of point-by-point or line-by-line relaxation is important for efficient convergence and it can approximately be determined by the coefficients of the EPDEs. However, experimentation might be required in some cases. Detailed information on this package is found in Adams (1989, 1991) and Adams and Smolarkiewicz (2001).

In general, atmospheric models use very fine resolution in terms of height or hydrostatic pressure in the lower troposphere. In this model, uniform vertical, as well as horizontal, grid spacing is required because it is a feature of the multigrid solver. To achieve the constant grid spacing with nonuniform height intervals in the vertical direction, one of the possibilities is to apply a monotonic function to general terrain-following coordinates, similar to that done in the Canadian Mesoscale Compressible Community model (Thomas et al. 1998) and the German Deutscher Wetterdienst (DWD) Lokal-Modell (Thomas et al. 2000). Here, we propose a flexible hybrid coordinate, which is also terrain following and has the features of the (modified) hybrid coordinate (Arakawa and Lamb 1977; Simmons and Burridge 1981; Simmons and Strüfing 1981), to meet the requirement of constant grid spacing. The flexible hybrid coordinate is briefly introduced in section 2. The description of the nonhydrostatic model is given in section 3 and the emphasis is put on solving the high-frequency wave equations. In section 4, three experiments are carried out to validate the model performance and to test the efficiency and accuracy of the multigrid method. Finally, a brief summary is given in section 5.

2. Flexible hybrid coordinate

Either pressure or height can be used to define the flexible hybrid coordinate (FHC), σ_f . Here, we only present $\sigma_f(z)$, which is the one used in this model. In physical space, z at any grid point is split into three parts: the surface height (z_s), base height (z_b), and deviation height (z_d). The implicit relationship between $\sigma_f(z)$ and z can be expressed as

$$z = z_s + \underbrace{F_b(\sigma_f)(z_t - z_{s,\max})}_{z_b} + \underbrace{F_d(\sigma_f)(z_{s,\max} - z_s)}_{z_d}, \quad (1)$$

where z_t is the height at the model top, and $z_{s,\max}$ is the maximum surface height within the domain; F_b and F_d are termed base and deviation functions, respectively; σ_f has the value 1 at the surface and then monotonically decreases with height to 0 at the model top, while the base (F_b) and deviation (F_d) functions are 0 at the surface and 1 at the model top according to their definitions here. The definitions of F_b and F_d depend upon the preferred vertical structure.

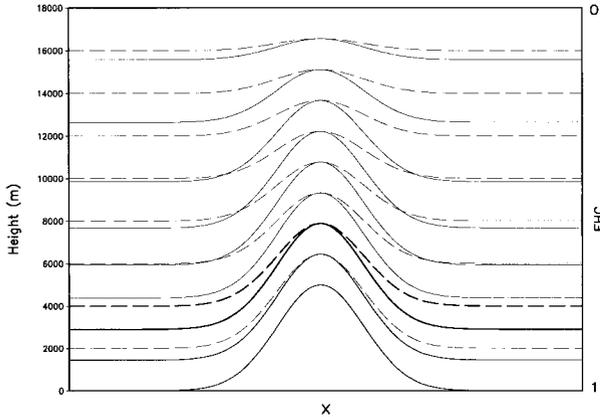


FIG. 3. The vertical cross section of σ_f surfaces with a linear deviation function $F_d = 1 - \sigma_f$ (dashed lines), and a hyperbolic tangent deviation function $F_d = [\tanh(X) - \tanh(C_1)]/[\tanh(C_2) - \tanh(C_1)]$ (solid lines), where $C_1 = -4$, $C_2 = 1$, and $X = C_2 - (C_2 - C_1)\sigma_f$. A linear base function $F_d = 1 - \sigma_f$ is used. The thick lines have the same value of σ_f .

The base function is mainly designed to meet the requirement of the constant vertical interval in the multigrad solver. A segment of any function can be chosen to present F_b , as well as F_d , in $\sigma_f(z)$ as long as it is well defined, satisfies boundary conditions, and monotonically decreases with σ_f . Suggested functions are hyperbolic tangent, polynomial, and exponential functions.

The deviation height could be used to assist the adjustment of the vertical resolution when the surface topography [or pressure in $\sigma_f(p)$] varies significantly. Figure 3 shows the vertical cross section of σ_f surfaces when two different deviation functions are used with the same linear base function. Both σ_f 's have the same resolution at the column of mountain peak, but the hyperbolic tangent deviation function (solid lines) gives a better resolution in terms of height at the valleys than the linear one (dashed lines) does. The difference of the resolutions can also be seen from the thick solid and dashed lines (Fig. 3), which present the same σ_f surface of 0.78. This higher resolution in the lower troposphere at the valleys due to the use of a hyperbolic tangent deviation function might be able to improve some simulations of boundary layer phenomena. It is noted that the usual σ - z coordinate $(z_i - z)/(z_i - z_s)$ is a special case of $\sigma_f(z)$ when $F_b = F_d = 1 - \sigma_f$ (dashed lines in Fig. 3). Therefore, the deviation function in FHC does provide more flexibility than the usual σ coordinate does.

A brief description of how to choose different coefficients in the hyperbolic tangent function, as in the formula in the caption of Fig. 3, might be useful. Here, C_1 ($\sigma_f = 1$) and C_2 ($\sigma_f = 0$) are the starting and ending points of the chosen segment $\tanh(X)$. The segment can be used to define the base and deviation functions after it is shifted downward for $\tanh(C_1)$ [minus $\tanh(C_1)$] and then normalized by the value of $\tanh(C_2) -$

$\tanh(C_1)$. The resolution in terms of height is related to the slope of the segment. A smaller slope ($\partial F_b/\partial X$ or $\partial F_d/\partial X$) indicates a higher resolution. As the coefficients ($C_1 = -4$ and $C_2 = 1$) used in Fig. 3, a higher resolution is obtained at the lower troposphere due to a smaller slope close to the starting point of C_1 . To experiment with different coefficients is recommended in order to get a better sense of this coordinate.

FHC is named not only for its flexibility, but also for the feature it has in common with the (modified) hybrid coordinate (Simmons and Burridge 1981; Simmons and Strüfing 1981) when $\sigma_f(p)$ is used, that is, that σ_f surfaces are (nearly) parallel to pressure surfaces in the stratosphere if the deviation function, F_d , is chosen properly (Chen 1999).

3. Model description

a. Governing equations

The basic governing equations of this fully compressible three-dimensional nonhydrostatic model in Cartesian coordinates are as follows:

$$\frac{\partial u}{\partial t} = -C_p \theta m_u \frac{\partial \pi}{\partial x} - \mathbf{V} \cdot \nabla u + D_u, \quad (2)$$

$$\frac{\partial v}{\partial t} = -C_p \theta m_v \frac{\partial \pi}{\partial y} - \mathbf{V} \cdot \nabla v + D_v, \quad (3)$$

$$\frac{\partial w}{\partial t} = -C_p \theta \frac{\partial \pi}{\partial z} - g - \mathbf{V} \cdot \nabla w + D_w, \quad (4)$$

$$\frac{\partial \pi}{\partial t} = -\frac{R}{C_v} \pi \nabla \cdot \mathbf{V} - \mathbf{V} \cdot \nabla \pi, \quad (5)$$

$$\frac{\partial \theta}{\partial t} = -\mathbf{V} \cdot \nabla \theta + D_\theta, \quad (6)$$

where Eqs. (2)–(6) are the x , y , and z components of the momentum equations, conservation of mass, and thermodynamic equation. The symbols u , v , w , π , and θ are velocity components in the x , y , and z directions; Exner function; and potential temperature, respectively. The Exner function, π , is defined as

$$\pi = \left(\frac{p}{p_0} \right)^{R/C_p},$$

where p is pressure and $p_0 = 1000$ hPa. In addition, C_p is the specific heat of air at constant pressure, R the gas constant, \mathbf{V} the three-dimensional velocity, m_u the mapping factor at the u point, m_v the mapping factor at the v point, and g the gravitational acceleration.

As mentioned earlier, FHC in terms of height, $\sigma_f(z)$, is applied to this model. The base function $F_b(\sigma_f)$ and deviation function $F_d(\sigma_f)$ are chosen case by case in

TABLE 1. Sixteen runs (four groups) with different base functions (F_b), grid intervals, tolerance numbers, time steps, and relaxation choices are set up for the experiment of linear mountain waves. In group 1, the formula of the hyperbolic tangent base function F_b (tanh) is the same as the hyperbolic tangent deviation function used in Fig. 3 except $C_1 = -2.2$ and $C_2 = 0.1$. An exponential function, $F_b(\text{exp}) = \exp[\ln 2 \times (1 - \sigma_f)] - 1$, is used in group 2. A linear base function is used in both group 3 and 4. In column 3 “point” and “line” indicate the choices of point-by-point and line-by-line relaxations, respectively. Same domain and the V(1, 1) multigrid cycle are used for all runs.

| Group | Runs | Relaxation | Tolerance number | F_b | Grid intervals | | |
|-------|------------|------------|------------------|----------------|----------------|------------|------------|
| | | | | | x | σ_f | Δt |
| 1 | TANH_P_1 | Point | 0.1 | f_b (tanh) | 256 | 160 | 10 s |
| | TANH_L_1 | Line | | | | | |
| | TANH_P_001 | Point | 0.001 | | | | |
| | TANH_L_001 | Line | | | | | |
| 2 | EXP_P_1 | Point | 0.1 | F_b (exp) | 256 | 160 | 10 s |
| | EXP_L_1 | Line | | | | | |
| | EXP_P_001 | Point | 0.001 | | | | |
| | EXP_L_001 | Line | | | | | |
| 3 | LIN_P_1 | Point | 0.1 | $1 - \sigma_f$ | 256 | 160 | 10 s |
| | LIN_L_1 | Line | | | | | |
| | LIN_P_001 | Point | 0.001 | | | | |
| | LIN_L_001 | Line | | | | | |
| 4 | LINH_P_1 | Point | 0.1 | $1 - \sigma_f$ | 512 | 320 | 5 s |
| | LINH_L_1 | Line | | | | | |
| | LINH_P_001 | Point | 0.001 | | | | |
| | LINH_L_001 | Line | | | | | |

order to meet the desired physical vertical resolutions in section 4. The potential temperature (θ), Exner function (π), and pressure (p) are divided into mean fields [$\bar{\theta}(z)$, $\bar{\pi}(z)$, and $\bar{p}(z)$] and perturbation fields [$\theta'(x, y, z, t)$, $\pi'(x, y, z, t)$, and $p'(x, y, z, t)$]. Mean fields are assumed hydrostatic balance, and $\bar{\pi}$ is expressed as follows:

$$\bar{\pi} = \left(\frac{\bar{p}}{p_o} \right)^{R/C_p}$$

The mean temperature (\bar{T}) is calculated from $\bar{\theta}$ and $\bar{\pi}$ using a Poisson equation, and it is also a function of height. After substituting mean and perturbation fields into Eqs. (2)–(6), the mean pressure gradient forces disappear (the vertical one is canceled by the gravitational force); $\partial\pi/\partial t$ becomes $\partial\pi'/\partial t$ in (5) and $\partial\theta/\partial t$ becomes $\partial\theta'/\partial t$ in (6).

b. Numerical methods

In order to relax the small time step constraint due to high-frequency acoustic and gravity waves, the semi-implicit scheme (Robert 1969) is applied. The low-frequency waves are calculated explicitly, while the high-frequency waves are calculated implicitly. In this section, most attention is paid to solving high-frequency wave equations. A staggered Arakawa C–Lorenz grid is used with the pressure and temperature located at the same levels as horizontal velocity components.

To solve the governing equations, we first advance prognostic variables with advection terms using a fourth-order advection scheme (Sun 1993) in the horizontal direction and a second-order Crowley advection scheme (Crowley 1968) in the vertical direction. For an

open boundary condition, the upstream method and the second-order Crowley scheme are applied to lateral boundaries, and the upstream method is also applied to the upper boundary. A free-slip boundary condition is imposed at the lower boundary. Second, the prognostic variables are advanced with eddy diffusion terms explicitly based on Durran and Klemp (1983).

Finally, the pressure gradient forces and divergence are evaluated. By the chain rule, the partial differentiation relationships between Cartesian coordinates and flexible hybrid coordinates are as follows:

$$\begin{aligned} \left. \frac{\partial}{\partial x} \right|_z &= \left. \frac{\partial}{\partial x} \right|_{\sigma_f} - \frac{\partial \sigma_f}{\partial x} \frac{\partial}{\partial \sigma_f} = \left. \frac{\partial}{\partial x} \right|_{\sigma_f} - B_x \frac{\partial}{\partial \sigma_f}, \\ \left. \frac{\partial}{\partial y} \right|_z &= \left. \frac{\partial}{\partial y} \right|_{\sigma_f} - \frac{\partial \sigma_f}{\partial y} \frac{\partial}{\partial \sigma_f} = \left. \frac{\partial}{\partial y} \right|_{\sigma_f} - B_y \frac{\partial}{\partial \sigma_f}, \quad \text{and} \\ \left. \frac{\partial}{\partial z} \right| &= \frac{\partial \sigma_f}{\partial z} \frac{\partial}{\partial \sigma_f} = B_z \frac{\partial}{\partial \sigma_f}, \end{aligned}$$

where $B_x = \partial\sigma_f/\partial x$, $B_y = \partial\sigma_f/\partial y$, and $B_z = \partial\sigma_f/\partial z$. After transferring from Cartesian coordinates to flexible hybrid coordinates, the equations for high-frequency waves become

$$\begin{aligned} \frac{\partial u}{\partial t} + C_p \bar{\theta} m_u \frac{\partial \pi'}{\partial x} - C_p \bar{\theta} m_u B_x \frac{\partial \pi'}{\partial \sigma_f} \\ = -C_p \theta' m_u \frac{\partial \pi'}{\partial x} + C_p \theta' m_u B_x \frac{\partial \pi'}{\partial \sigma_f}, \end{aligned} \tag{7}$$

$$\begin{aligned} \frac{\partial v}{\partial t} + C_p \bar{\theta} m_v \frac{\partial \pi'}{\partial y} - C_p \bar{\theta} m_v B_y \frac{\partial \pi'}{\partial \sigma_f} \\ = -C_p \theta' m_v \frac{\partial \pi'}{\partial y} + C_p \theta' m_v B_y \frac{\partial \pi'}{\partial \sigma_f}, \end{aligned} \tag{8}$$

$$\frac{\partial w}{\partial t} + C_p \bar{\theta} B_z \frac{\partial \pi'}{\partial \sigma_f} - \frac{\theta'}{\theta} g = -C_p \theta' B_z \frac{\partial \bar{\pi}}{\partial \sigma_f}, \quad (9)$$

$$\frac{\partial \pi'}{\partial t} + \frac{R}{C_v} \bar{\pi} \nabla \cdot \mathbf{V} = -\frac{R}{C_v} \pi' \nabla \cdot \mathbf{V}, \quad (10)$$

$$\frac{\partial \theta'}{\partial t} + w B_z \frac{\partial \bar{\theta}}{\partial \sigma_f} = 0. \quad (11)$$

Only terms on the left-hand sides of (7)–(11) are integrated implicitly, while terms on the right-hand sides are evaluated explicitly. An implicit scheme with uncentered coefficient α (Cullen 1990; Juang 1992) is applied. For convenience, we define any prognostic variable, M , such that M^n is the variable at the current time step n , M^* is the updated variable after the integration of the eddy diffusion, M^{**} is the temporary prognostic variable for integrating the high-frequency waves, and M^{n+1} is the variable at the next time step $n + 1$. In order to save memory, only the Exner function perturbation is expressed as

$$\pi' = \alpha \pi'^{**} + (1 - \alpha) \pi'^n = \pi'^\alpha.$$

Other predictors, M , are expressed as

$$M = \alpha M^{**} + (1 - \alpha) M^*,$$

since the differences between M^* and M^n are small. Here, $0.5 < \alpha \leq 1$ is used to guarantee unconditional stability and a test of the α value is examined in section 4.

After applying the implicit scheme to the left-hand-side terms of (7)–(11) and discretizing time derivatives, we obtain

$$\frac{u^{**}}{m_u} = \frac{u^*}{m_u} - \Delta t C_p \bar{\theta} \frac{\partial \pi'^\alpha}{\partial x} + \Delta t C_p \bar{\theta} B_x \frac{\partial \pi'^\alpha}{\partial \sigma_f}, \quad (12)$$

$$\frac{v^{**}}{m_v} = \frac{v^*}{m_v} - \Delta t C_p \bar{\theta} \frac{\partial \pi'^\alpha}{\partial y} + \Delta t C_p \bar{\theta} B_y \frac{\partial \pi'^\alpha}{\partial \sigma_f}, \quad (13)$$

$$w^{**} = w^* - \Delta t C_p \bar{\theta} B_z \frac{\partial \pi'^\alpha}{\partial \sigma_f} + \Delta t \alpha \frac{\theta'^{**}}{\theta} g + \Delta t (1 - \alpha) \frac{\theta'^*}{\theta} g, \quad (14)$$

$$\begin{aligned} \pi'^{***} = & \pi'^* - \frac{\Delta t R}{C_v} \bar{\pi} \\ & \times \left[m_p^2 \alpha \frac{\partial u^{**}/m_u}{\partial x} + m_p^2 (1 - \alpha) \frac{\partial u^*/m_u}{\partial x} \right. \\ & - m_p^2 \alpha B_x \frac{\partial u^{**}/m_u}{\partial \sigma_f} - m_p^2 (1 - \alpha) B_x \frac{\partial u^*/m_u}{\partial \sigma_f} \\ & + m_p^2 \alpha \frac{\partial v^{**}/m_v}{\partial y} + m_p^2 (1 - \alpha) \frac{\partial v^*/m_v}{\partial y} \\ & - m_p^2 \alpha B_y \frac{\partial v^{**}/m_v}{\partial \sigma_f} - m_p^2 (1 - \alpha) B_y \frac{\partial v^*/m_v}{\partial \sigma_f} \\ & \left. + \alpha B_z \frac{\partial w^{**}}{\partial \sigma_f} + (1 - \alpha) B_z \frac{\partial w^*}{\partial \sigma_f} \right], \quad (15) \end{aligned}$$

$$\begin{aligned} \theta'^{***} = & \theta'^* - \Delta t \alpha w^{**} B_z \frac{\partial \bar{\theta}}{\partial \sigma_f} \\ & - \Delta t (1 - \alpha) w^* B_z \frac{\partial \bar{\theta}}{\partial \sigma_f}, \quad (16) \end{aligned}$$

where m_p is the mapping factor at the point of pressure. In addition, w^{**} can be obtained from (14) and (16) and expressed as

$$w^{**} = \frac{w^* \left[1 - \Delta t^2 \alpha (1 - \alpha) \frac{g}{\theta_v} B_z \frac{\partial \bar{\theta}}{\partial \sigma_f} \right] - \Delta t C_p \bar{\theta} B_z \frac{\partial \pi'^\alpha}{\partial \sigma_f} + \Delta t g \frac{\theta'^*}{\theta}}{E}, \quad (17)$$

where

$$E = 1 + \Delta t^2 g (\alpha^2 / \bar{\theta}) B_z (\partial \bar{\theta} / \partial \sigma_f).$$

The calculation of w^{**} should involve the calculation of the inverse of a tridiagonal matrix if the space is discretized. Here, we keep the continuous form in space to simplify the calculation of w^{**} , which will reduce the complexity of the EPDE of the Exner function perturbation later on. In Thomas et al. (2000), the Arakawa C–Lorenz grid was also implemented, but they used temperature, pressure, and vertical momentum equations to calculate w without mass lumping.

After substituting (12), (13), and (17) into (15) and reorganizing terms, we can obtain an EPDE with two

cross-derivative terms for the linear part of high-frequency waves. The final form of the EPDE is

$$\begin{aligned} C_{xx} \frac{\partial^2 \pi'^{**}}{\partial x^2} + C_{yy} \frac{\partial^2 \pi'^{**}}{\partial y^2} + C_{zz} \frac{\partial^2 \pi'^{**}}{\partial \sigma_f^2} + C_{xz} \frac{\partial^2 \pi'^{**}}{\partial x \partial \sigma_f} \\ + C_{yz} \frac{\partial^2 \pi'^{**}}{\partial y \partial \sigma_f} + C_x \frac{\partial \pi'^{**}}{\partial x} + C_y \frac{\partial \pi'^{**}}{\partial y} + C_z \frac{\partial \pi'^{**}}{\partial \sigma_f} \\ + C_e \pi'^{**} = r, \quad (18) \end{aligned}$$

where the formulas of the coefficients C_{xx} , C_{yy} , C_{zz} , C_{xz} , C_{yz} , C_x , C_y , C_z , and C_e , and the source/sink term r , are given in appendix B. The C coefficients on the left-hand side of (18) are independent of time and, therefore, only

need to be calculated once. However, extra memory is required to save these coefficients.

It is worth mentioning again that the cross-derivative terms are kept on the left-hand side of (18) to ensure that the model is more stable in steep-terrain simulations. Only very few fully compressible nonhydrostatic models have been done in a similar way (Skamarock et al. 1997; Thomas et al. 1998, 2000). Equation (18) represents one point, and in general, more than 300 000 points are used in a three-dimensional nonhydrostatic model. Therefore, it is necessary to solve a set of linear equations with a matrix of ($\sim 300\,000 \times 300\,000$), and the multigrid method is chosen to solve the EPDEs. Instead of writing a new multigrid solver, the multigrid solver mud3cr, which has second-order accuracy and is restricted to the finite-difference operators, is applied. The C coefficients and r term in (18) are calculated and passed into the multigrid solver with the first guess of π'^n . The iteration is stopped when the maximum outer iteration number is reached or when $r_{\max} < \text{tol}$, where tol is a given error tolerance number, and r_{\max} is the convergence criteria, which is defined as

$$r_{\max} = \frac{\|\pi'^{**}(K) - \pi'^{**}(K-1)\|}{\|\pi'^{**}(K)\|},$$

where K and $K-1$ indicate the current and previous iteration multigrid cycles, respectively, and $\|\cdot\|$ is the maximum vector norm and is used to compute the relative difference between two successive solutions. After getting π'^{**} , we can substitute it into (12), (13), and (17) to calculate u^{**} , v^{**} , and w^{**} , respectively. The prognostic variables at the next time step (u^{n+1} , v^{n+1} , w^{n+1} , and π'^{n+1}) are finally obtained after we explicitly integrate terms on the right-hand sides of equations (7)–(10).

c. Boundary conditions of the Exner function of the EPDEs

The condition $\pi'^{**} = 0$ is applied to the top boundary and lateral boundaries in the x direction. For the boundary condition in the y direction, $\pi'^{**} = 0$ is used if the experiment is three-dimensional, and a periodic boundary condition is used if the experiment is two-dimensional (y direction is assumed homogeneous). At the lower boundary, the velocity normal to the surface is equal to zero, and therefore,

$$u^{**}m_p B_x + v^{**}m_p B_y + w^{**}B_z = 0. \quad (19)$$

As mentioned earlier, the Arakawa C–Lorenz grid is used with the pressure, temperature, and horizontal velocity components located at the same levels. However, in order to satisfy the lower boundary condition, all prognostic variables are located on the surface (horizontal still staggered). The grid spacing is uniform between π'^{**} , and w is located between π'^{**} . Therefore, the grid interval between w is not a constant (only half-grid interval at the lower boundary). Similar grid points

are also arranged at upper and lateral boundaries. Substituting (12), (13), and (17) into (19) yields

$$\frac{\partial \pi'^{**}}{\partial \sigma_f} - \frac{m_u m_p B_x}{D} \frac{\partial \pi'^{**}}{\partial x} - \frac{m_v m_p B_y}{D} \frac{\partial \pi'^{**}}{\partial y} = f_1, \quad (20)$$

where

$$f_1 = -\frac{1}{\Delta t C_p \bar{\theta} D \alpha} \left\{ m_p (B_x u^* + B_y v^*) - B_z E^{-1} \left[1 - \Delta t^2 \alpha (1 - \alpha) \frac{g}{\theta} B_z \frac{\partial \bar{\theta}}{\partial \sigma_f} \right] w^* - E^{-1} \Delta t g B_z \frac{\theta'^*}{\theta} \right\} - \frac{(1 - \alpha)}{\alpha} \left(\frac{\partial \pi'^n}{\partial \sigma_f} - \frac{m_u m_p B_x}{D} \frac{\partial \pi'^n}{\partial x} - \frac{m_v m_p B_y}{D} \frac{\partial \pi'^n}{\partial y} \right),$$

and

$$D = m_u m_p B_x^2 + m_v m_p B_y^2 + E^{-1} B_z^2.$$

In practice, Eq. (20) is approximately solved as

$$\frac{\partial \pi'^{**}}{\partial \sigma_f} = f_1 + \frac{m_u m_p B_x}{D} \frac{\partial \pi'^*}{\partial x} + \frac{m_v m_p B_y}{D} \frac{\partial \pi'^*}{\partial y}.$$

4. Experiments

Three experiments, linear mountain waves (Queney 1948), a downslope windstorm (Lilly 1978), and a three-dimensional thermal bubble, are carried out to demonstrate the performance of the model and the efficiency and accuracy of the multigrid solver. In order to compare the model results with previous works and with the analytical solution, the first two experiments are two-dimensional with the y direction homogeneous. The mapping factors (e.g., m_p , m_u , and m_v) are set to 1, the maximum outer iteration number is 30, and the uncentered coefficient, α , is 0.65 (for the reason explained later).

a. Linear mountain waves

A linear mountain wave experiment is chosen after Queney (1948). The nonlinear effects are suppressed when the mountain is very shallow. Therefore, the numerical results can be compared with the linear analytical solution. A bell-shaped ridge is imposed at the center of the domain. The ridge is defined as $h = h_s / [1 + (x/a)^2]$, where the crest of the ridge, h_s , is 10 m, and the half-width length, a , is 1000 m. The atmosphere is dry and the eddy diffusion is excluded. A uniform horizontal mean velocity, 10 m s^{-1} , is imposed for the whole atmosphere, which has a Brunt–Väisälä frequency (N) of 0.01 s^{-1} . The sea level pressure, vertical domain, horizontal domain in the x direction, and

surface temperature are 1000 hPa, 25 600 m, 51 200 m, and 300 K, respectively. How to properly handle the open boundary condition is an important issue, but it is not the primary concern of this study. Therefore, we only simply apply the boundary conditions, which were mentioned in section 3c, with sponge zones (30 bands in the x lateral boundaries and one-fourth domain at upper boundary) to absorb waves. Sixteen runs (four groups) are set up, as indicated in Table 1, with the use of a $V(1,1)$ multigrad cycle. A linear deviation function is applied to all runs. Different base functions and different numbers of grid intervals are examined in order to see the influence of the resolution on the surface pressure perturbation, as well as in the lower troposphere. Moreover, the convergence behavior of the multigrad method regarding different tolerance numbers, and relaxation choices (point by point or line by line) are tested. There are four grid intervals in the y direction, and $\Delta x = \Delta y$ is set in every run. A hyperbolic tangent base function as described in the caption of Table 1 is used in group 1. A very high resolution (~ 20 m) is located near the surface and a relatively low resolution (~ 420 m) is presented near the top. The exponential base function, which is also given in the caption of Table 1, is utilized in group 2. The vertical resolution is about 140 m near the surface and 270 m near the top. A linear base function ($1 - \sigma_p$) is applied to the rest. The last group has double grid intervals in both the x and σ_f directions compared with the others. Therefore, the average vertical resolution over the whole domain is about 200 m in group 3, and about 100 m in group 4. A 200-m horizontal resolution and a 10-s time step are used in the first three groups; a 100-m horizontal resolution and a 5-s time step are used in the last one. The model integrates for 10 h at which time the solution approximately reaches steady state.

Figure 4a shows the surface pressure perturbation for the analytical solution of the nonhydrostatic mountain waves (ANALY), and the model simulations of TANH.P.1, EXP.P.1, LIN.P.1, and LINH.P.1 after a 10-h integration, and Fig. 4b shows the discrepancies of the surface pressure perturbation between the analytical solution and model simulations in Fig. 4a. Only one run of each group is plotted since four runs in the same group are very similar. It should be mentioned that the analytical solution in Queney (1948) is different from that in Fig. 4a, which is calculated with higher-order accuracy (Hsu and Sun 2001). Weak high pressure (~ 0.5 Pa) is built on the windward side of the ridge since air is accumulated there and a minimum low pressure center is located on the crest of the ridge as indicated in Fig. 4a. Regarding the phases of the surface pressure perturbations, the model simulations are quite consistent with the analytical solution for all runs, although most of the amplitudes are underestimated. The root-mean-square errors of the surface pressure perturbation in Table 2, as well as the plot of Fig. 4b, show that group 1 is in the best agreement with the analytical

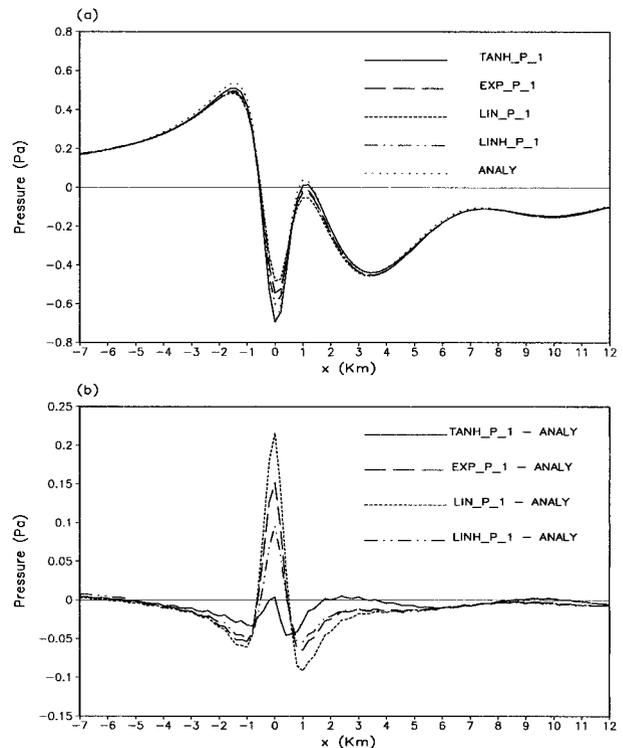


FIG. 4. (a) The surface pressure perturbation for the analytical solution (ANALY) (dotted lines) and the model simulations of TANH.P.1 (solid line), EXP.P.1 (long-dashed line), LIN.P.1 (short-dashed line), and LINH.P.1 (dashed-double-dot line) after a 10-h integration. (b) The results in (a) are used to plot the errors of the surface pressure perturbation of the model simulations.

solution, followed by the results of group 4, group 2, and group 3, in that order. The simulated result of the surface pressure perturbation is strongly correlated to the vertical resolution near the ground.

Figure 5 shows the potential temperature perturbation for the analytical solution (dashed lines) and the model simulation results (solid lines). The overall patterns of the model simulations are quite comparable with the analytical solution. For the model results, the amplitudes are slightly underestimated, and the phases are slightly shifted. Although the wave propagation from the surface is well simulated in group 1, the resolution in the lower troposphere is inhomogeneous and not sufficient. Thus, it does not produce a better result (Fig. 5a and Table 2) compared with group 4 (Fig. 5d and Table 2), which has double grid intervals in each direction and a higher time resolution. The root-mean-square errors of groups 3 and 4 in Table 2 indicate that the improvement in the surface pressure (improved 46%) is faster than the potential temperature (improved 33%) as resolution increases, namely that the solution of the surface pressure perturbation, which is obtained by solving EPDEs, is more sensitive than the solution of the potential temperature perturbation. It is noted that the total central processing unit (CPU) times in group 4 are much more

TABLE 2. The average iteration numbers in terms of the multigrid V cycles per time step are presented. The root-mean-square errors of the surface pressure perturbation (p'_s) and the potential temperature perturbation (θ') are calculated within the plotting domain of Figs. 4 and 5, respectively.

| Group | Runs | Average iteration no. of V (1, 1) cycles per time step | CPU time (h) | Root-mean-square error | |
|-------|------------|---|-----------------|------------------------|------------------------|
| | | | | p'_s (Pa) | θ' (K) |
| 1 | TANH_P_1 | 1.020 | 2.06 | 0.130×10^{-1} | 0.153×10^{-2} |
| | TANH_L_1 | 1.073 | 2.33 | 0.129×10^{-1} | 0.153×10^{-2} |
| | TANH_P_001 | 13.115 | 7.04 | 0.131×10^{-1} | 0.153×10^{-2} |
| | TANH_L_001 | 6.596 | 7.05 | 0.129×10^{-1} | 0.153×10^{-2} |
| 2 | EXP_P_1 | 1.023 | 2.18 | 0.315×10^{-1} | 0.166×10^{-2} |
| | EXP_L_1 | 1.021 | 2.35 | 0.311×10^{-1} | 0.166×10^{-2} |
| | EXP_P_001 | 6.337 | 4.64 | 0.304×10^{-1} | 0.166×10^{-2} |
| | EXP_L_001 | 6.131 | 6.71 | 0.304×10^{-1} | 0.166×10^{-2} |
| 3 | LIN_P_1 | 1.024 | 2.06 | 0.435×10^{-1} | 0.172×10^{-2} |
| | LIN_L_1 | 1.023 | 2.30 | 0.432×10^{-1} | 0.172×10^{-2} |
| | LIN_P_001 | 6.011 | 4.12 | 0.424×10^{-1} | 0.172×10^{-2} |
| | LIN_L_001 | 6.253 | 6.84 | 0.424×10^{-1} | 0.172×10^{-2} |
| 4 | LINH_P_1 | 1.019 | 19.74 | 0.233×10^{-1} | 0.115×10^{-2} |
| | LINH_L_1 | 1.019 | 25.45 | 0.223×10^{-1} | 0.115×10^{-2} |
| | LINH_P_001 | 8.091 | 44.70 | 0.227×10^{-1} | 0.114×10^{-2} |
| | LINH_L_001 | 7.906 | 92.40 | 0.227×10^{-1} | 0.114×10^{-2} |

than four times those in group 3 when only four times grid intervals are used in group 4. We suspect that this performance anomaly is due to virtual memory effects. The initial case was designed to use most of the physical memory on the processor. The runs in group 4 did not fit into physical memory and required paging to and from the disk.

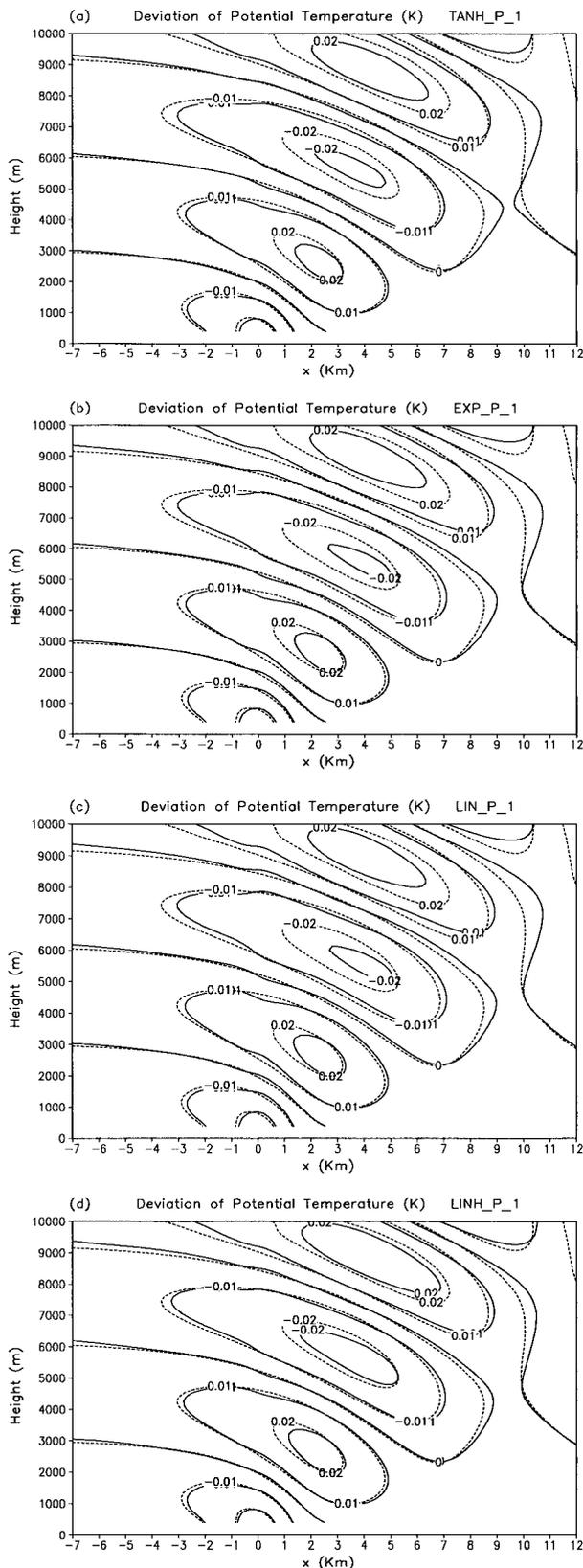
The tolerance number is one of the major factors, which determines the convergence rate and the accuracy of the solution. Two tolerance numbers, 0.1 and 0.001, are tested, and the model simulation errors are almost identical for both numbers as indicated in Table 2 (e.g., TANH_P_1 vs TANH_P_001, TANH_L_1 vs TANH_L_001, etc.). However, the average iteration numbers, which are defined as the average numbers of V cycles per time step, are quite different. The solutions are converged with one average V(1,1) cycle when the 0.1 tolerance number is used. Without further improving accuracy, it takes 6–13 average V(1,1) cycles to converge, depending on the different base functions and relaxation choices, when the 0.001 tolerance number is used. The tolerance number of 0.1 apparently works quite well in this linear case.

The relaxation choice, point by point or line by line, could also critically determine the convergence rate, and the decision depends upon the coefficients of the EPDEs. Details are found in Fulton et al. (1986). The point-by-point relaxation with the tolerance number of 0.1 works quite well in this experiment as indicated in Table 2. For a V cycle, to solve the EPDEs line by line takes more time than to solve them point by point due to the involvement of calculating tridiagonal matrices in the line-by-line relaxation. Therefore, the total CPU time spent on the second run (line by line) is about 10%–30% more than that on the first run (point by point) for each group with the tolerance number of 0.1, though both iteration numbers are comparable. The grid boxes

in the lower troposphere are unisotropic ($\Delta z \ll \Delta x$) in group 1. When the tolerance number of 0.001 is chosen, the average iteration number with line-by-line relaxation is half of that with point-by-point relaxation, but both CPU times are comparable. As mentioned earlier, the results with a tolerance number of 0.1 are good enough in this case and the point-by-point relaxation is more efficient than the line-by-line one.

Another setup with the same configuration of TANH_P_1 but different sweeps of prerelaxation and postrelaxation—V(2,2), V(1,2), and V(2,1)—is tested. The root-mean-square errors of the surface pressure perturbation ($\sim 0.130 \times 10^{-1}$ Pa) and potential temperature perturbation ($\sim 0.153 \times 10^{-2}$ K) for these three runs are very similar to those in TANH_P_1. The average iteration number is about 1.019 for all of them, and the total CPU time for each of them is slightly more than that for TANH_P_1, which uses a V(1,1) cycle. So far, the V(1,1) cycle with point-by-point relaxation has performed reasonably well in this nonhydrostatic linear mountain wave case.

The uncentered coefficient, α , determines the instability and the damping strength of the short waves. Therefore, its value might be one of the critical factors in determining the model accuracy and the convergence rate of the EPDEs. When the time derivative is discretized using three time steps, the short waves are stable if $0.5 < \alpha \leq 1$ (Cullen 1990). The waves damp the most as α equals 1 (fully implicit scheme). However, when a two-time-step scheme is applied, the proof of the instability regarding the value of α is not that straightforward (Simmons and Temperton 1996), but its value is still within the same range (0.5–1). In Juang (2000) and Ikawa (1988), $\alpha = 0.8$ was chosen. In this study, instead of choosing an arbitrary number, a test with different values of α is conducted with the same



configuration of TANHLP_1. Figure 6 shows the results of the relative root-mean-square error and the average iteration number. The root-mean-square error of the potential temperature perturbation increases as α increases. The error of the surface pressure perturbation behaves quite similarly when α is greater than 0.60, but the values rise as α approaches 0.5. With the value of 0.52, the error grows one order larger in the potential temperature perturbation and three orders larger in the surface pressure perturbation. The model becomes unstable when $\alpha \leq 0.51$. The convergence rate is correlated to the value of α , which indicates the damping strength of the short waves or the smoothness of short waves. The average iteration number drops fast at the beginning as α increases and then approaches a value of 1 when α is greater than 0.65. Considering the accuracy of the pressure field and the convergence efficiency, $\alpha = 0.65$ is, therefore, adopted into this model.

A hydrostatic mountain wave test is also given after Queney (1948). The initial conditions of the atmosphere, the model setup, and the options in the multigrid solver are the same as those in TANHLP_1, except that the half-width length of the mountain is 10 km, the horizontal domain is 640 000 m, the horizontal resolution is 2000 m, and the time interval is 20 s. The grid aspect ratio ($\Delta x/\Delta z$) is about 100 in the lower troposphere. Different tolerance numbers and relaxation choices are tested, and the results are shown in Table 3. The CPU times in Table 3 are less than those in TANHLP_1 since a larger time step is used. The root-mean-square errors of the surface pressure perturbation with point-by-point relaxation (P_1 and P_001) are slightly better than those with line-by-line relaxation (L_1 and L_001), but all of them are still considered to converge to similar solutions. With similar accuracy, the point-by-point relaxation with the tolerance number of 0.1 (P_1) is still the most efficient—the same conclusion as in the nonhydrostatic mountain wave simulation.

b. Downslope windstorm

Strong downslope winds, which develop on the steep lee sides of mountains, have received a great deal of attention (e.g., Lilly 1978; Durran and Klemp 1983; Smith 1985; Richard et al. 1990; Miller and Durran 1991; Doyle et al. 2000; Hsu and Sun 2001). To further demonstrate the nonlinear behavior of this model and to test the efficiency of the multigrid solver, we choose to simulate the hydraulic jump (Fig. 7 in Lilly 1978) and the strong downslope wind (Fig. 9 in Lilly 1978) that occurred during the windstorm experiment on 11

FIG. 5. The potential temperature perturbations (K) for the analytical solution (dashed lines) and the model simulation results (solid lines) of (a) TANHLP_1, (b) EXP_P_1, (c) LIN_P_1, and (d) LINH_P_1 after a 10-h integration.

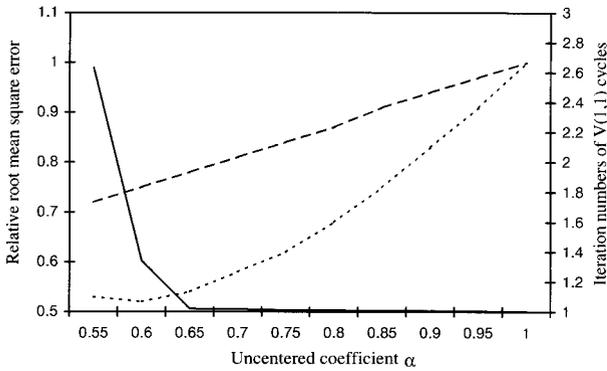


FIG. 6. The relative root-mean-square errors of the potential temperature perturbation (long-dashed line) and the surface pressure perturbation (short-dashed line) with different uncentered coefficients are plotted. The potential temperature perturbation is scaled by the value of 0.196×10^{-2} and the surface pressure perturbation is scaled by the value of 0.241×10^{-1} . Solid line shows the average iteration numbers in terms of V(1, 1) cycles.

January 1972 in Colorado. The isentropic surfaces at high levels were dragged down to low levels on the lee side of the mountain, and strong winds (greater than 60 m s^{-1}) developed near the surface. The sounding at 1200 UTC 11 January 1972 at Grand Junction, Colorado, is used as the initial condition (Table 4). As in the linear mountain wave experiment, the same boundary conditions are applied, and a bell-shaped mountain is also imposed into the domain with a crest of 2000 m and a half-width length of 10 km. The vertical domain is 35 km. A 30-band sponge zone is used at the top and both lateral boundaries in the x direction, and the horizontal resolution is 1 km. There are 384, 4, and 192 grid intervals in the x , y , and σ_f directions, respectively. Both linear base and deviation functions are the same as those in group 3 in the linear mountain wave experiment. The tolerance number is 0.1, the V(1,1) cycle is used, and the point-by-point relaxation is chosen. A 5-s time step is utilized and the model is integrated for 4 h.

Figures 7a and 7b show the solutions of the potential temperature at 2 and 3 h, respectively. The hydraulic jump is well formed after 2 h and continuously propagates downstream with a speed of about 4 m s^{-1} . The leading edge of the hydraulic jump becomes steeper and steeper as time goes on. The initial zonal wind near the

surface is about $8\text{--}10 \text{ m s}^{-1}$, but reaches a maximum of 62.9 m s^{-1} after 2 h of the simulation (Fig. 8a) and a maximum of 72.9 m s^{-1} after 3 h (Fig. 8b). The hydraulic jump and the magnitude of the downslope wind compare reasonably with observational data, as well as with most model results in Fig. 3 of Doyle et al. (2000). In this experiment, it takes about 2.61 average V(1,1) cycles to converge for point-by-point relaxation. The model takes one to two more V(1,1) cycles to converge at the very early stage of the simulations due to the initial shock, and thereafter only one V(1,1) cycle. However, when the hydraulic jump appears, the iteration number is back to three to four V(1,1) cycles. The average iteration number could be reduced to one average V(1,1) cycle when a 2-s time step is applied to the windstorm case. This feature is also pointed out in Skamarock et al. (1997), who used the conjugate residual method. Two other tests are also carried out: point-by-point relaxation with 0.001 tolerance number, and line-by-line relaxation with 0.1 tolerance number. Both results are quite comparable with those in the original test, which uses a 0.1 tolerance number and point-by-point relaxation. However, it takes 6.365 average V(1,1) cycles to converge with a tolerance number of 0.001 and point-by-point relaxation. The line-by-line relaxation with a 0.1 tolerance number takes only 1.51 average V(1,1) cycles per time step, but it requires more CPU time (5.26 h for point-by-point relaxation vs 6.37 h for line-by-line relaxation).

The tests mentioned above are carried out on an IBM RISC/6000 scalar machine with a 200-MHz processor and 4 MB of L2 cache. Since the field indices for both model and the multigrid solver mud3cr are (i, j, k) , the use of scalar machines might degrade the performance of line-by-line relaxation, while vector machines will not cause this problem. Therefore, another comparison between point-by-point relaxation and line-by-line relaxation with a tolerance number of 0.1 is carried out on a Cray J90 vector machine. The model CPU time with the point-by-point relaxation is 16.63 h. However, the CPU time with line-by-line relaxation is only 8.94 h.

As mentioned in section 2, the deviation function can be applied to adjust the vertical resolution in valley regions. Another setup with a hyperbolic tangent de-

TABLE 3. For the title of each run, P indicates point-by-point relaxation, L indicates line-by-line relaxation, 1 indicates the tolerance number of 0.1, and 001 indicates the tolerance number of 0.001. The surface pressure perturbation (p'_s) error is calculated within the horizontal domain of 120 000 m, and the potential temperature perturbation (θ') error is calculated within the domain of 120 000 m \times 10 000 m).

| Runs | Average iteration no. of V (1,1) cycles per time step | CPU time (h) | Root-mean-square error | |
|-------|---|--------------|------------------------|------------------------|
| | | | p'_s (Pa) | θ' (K) |
| P_1 | 1.096 | 1.39 | 0.129×10^{-1} | 0.490×10^{-2} |
| L_1 | 1.035 | 1.95 | 0.140×10^{-1} | 0.492×10^{-2} |
| P_001 | 5.048 | 3.27 | 0.129×10^{-1} | 0.492×10^{-2} |
| L_001 | 2.808 | 3.48 | 0.140×10^{-1} | 0.492×10^{-2} |

TABLE 4. The sounding, which was collected at 1200 UTC 11 Jan 1972 at Grand Junction, CO, and used for the downslope windstorm experiment.

| p (mb) | z (m) | T (°C) | RH (%) | u (m s ⁻¹) |
|----------|-----------|----------|--------|--------------------------|
| 850.00 | 1514.00 | -0.60 | 41.00 | 8.66 |
| 807.00 | 1929.00 | 0.20 | 36.00 | 8.66 |
| 800.00 | 1999.00 | -0.40 | 37.00 | 8.66 |
| 750.00 | 2510.00 | -5.00 | 46.00 | 8.66 |
| 700.00 | 3048.00 | -8.80 | 56.00 | 10.18 |
| 650.00 | 3618.00 | -13.10 | 73.00 | 13.52 |
| 624.00 | 3927.00 | -15.50 | 83.00 | 15.91 |
| 600.00 | 4222.00 | -17.50 | 79.00 | 17.00 |
| 595.00 | 4285.00 | -18.00 | 77.00 | 17.00 |
| 575.00 | 4540.00 | -18.40 | 75.00 | 20.97 |
| 550.00 | 4871.00 | -20.40 | 91.00 | 24.98 |
| 544.00 | 4952.00 | -20.90 | 95.00 | 25.98 |
| 528.00 | 5173.00 | -20.40 | 94.00 | 28.98 |
| 500.00 | 5574.00 | -22.90 | 75.00 | 30.70 |
| 475.00 | 5949.00 | -23.90 | 68.00 | 32.28 |
| 450.00 | 6342.00 | -26.40 | 64.00 | 33.96 |
| 432.00 | 6636.00 | -28.30 | 60.00 | 34.61 |
| 400.00 | 7182.00 | -33.50 | 71.00 | 36.53 |
| 353.00 | 8048.00 | -39.90 | 67.00 | 42.30 |
| 350.00 | 8109.00 | -40.40 | 990.00 | 42.30 |
| 300.00 | 9138.00 | -49.10 | 990.00 | 36.85 |
| 250.00 | 10 309.00 | -58.90 | 990.00 | 31.65 |
| 213.00 | 11 298.00 | -65.80 | 990.00 | 31.45 |
| 200.00 | 11 680.00 | -66.60 | 990.00 | 36.00 |
| 182.00 | 12 251.00 | -66.20 | 990.00 | 38.53 |
| 176.00 | 12 457.00 | -59.90 | 990.00 | 34.98 |
| 175.00 | 12 493.00 | -59.90 | 990.00 | 34.04 |
| 168.00 | 12 748.00 | -59.20 | 990.00 | 29.48 |
| 150.00 | 13 457.00 | -60.20 | 990.00 | 33.58 |
| 135.00 | 14 121.00 | -56.20 | 990.00 | 17.99 |
| 125.00 | 14 607.00 | -59.30 | 990.00 | 18.51 |
| 116.00 | 15 071.00 | -62.40 | 990.00 | 22.11 |
| 100.00 | 15 979.00 | -66.20 | 990.00 | 29.96 |
| 80.00 | 17 349.00 | -61.00 | 990.00 | 22.00 |
| 76.00 | 17 669.00 | -59.70 | 990.00 | 20.74 |
| 74.00 | 17 838.00 | -53.00 | 990.00 | 20.80 |
| 70.00 | 18 196.00 | -53.80 | 990.00 | 18.77 |
| 61.00 | 19 080.00 | -53.80 | 990.00 | 14.00 |
| 60.00 | 19 186.00 | -54.20 | 990.00 | 14.00 |
| 54.00 | 19 857.00 | -57.20 | 990.00 | 12.95 |
| 50.00 | 20 347.00 | -54.80 | 990.00 | 12.50 |
| 48.00 | 20 608.00 | -55.60 | 990.00 | 11.35 |
| 43.00 | 21 317.00 | -50.70 | 990.00 | 6.91 |
| 40.00 | 21 786.00 | -52.60 | 990.00 | 2.85 |
| 38.00 | 22 116.00 | -54.10 | 990.00 | 0.58 |
| 33.00 | 23 033.00 | -48.50 | 990.00 | 1.79 |
| 30.00 | 23 659.00 | -49.30 | 990.00 | 2.94 |
| 25.00 | 24 855.00 | -49.30 | 990.00 | 2.15 |
| 24.00 | 25 122.00 | -49.30 | 990.00 | 1.75 |
| 20.00 | 26 309.00 | -52.40 | 990.00 | 7.05 |
| 17.00 | 27 360.00 | -52.40 | 990.00 | 8.66 |
| 15.00 | 28 173.00 | -50.40 | 990.00 | 4.86 |

viation function is examined. Below 6 km, the hyperbolic tangent case has a near-constant vertical resolution of about 172 m, in contrast to the linear case where the lower-troposphere vertical resolution varies between 172 and 182 m. Figure 9 presents the potential temperature and zonal wind after a 4-h integration. We can see that the hydraulic jump in the hyperbolic tangent case (solid lines) moves faster than that in the linear case (dashed lines). The flexibility of the deviation func-

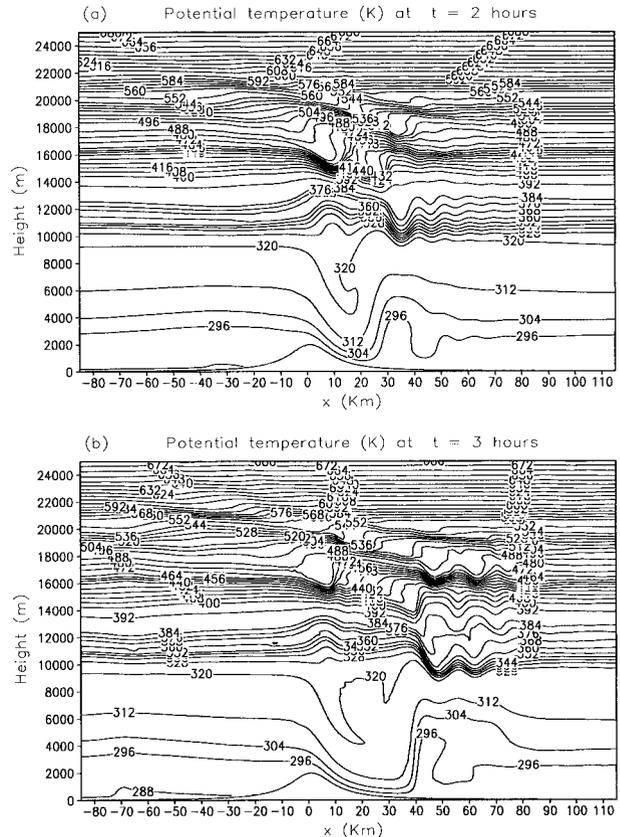


FIG. 7. The vertical cross sections of potential temperature (K) at (a) $t = 2$ h and (b) $t = 3$ h with the point-by-point relaxation and 0.1 tolerance number.

tion, which cannot be achieved by usual σ coordinates, provides a good opportunity to further study the influence of the resolution on the model simulations in valley regions. More investigation of the σ_f coordinate will be interesting, but it is beyond the scope of this paper.

c. A three-dimensional thermal bubble

The signals of the previous mountain wave experiments are mainly from the surface. Another experiment, using a three-dimensional thermal bubble with the forcing located in the lower troposphere, might be worth testing. In this experiment, the terrain is flat, and the atmosphere is calm and isentropic. A linear base function, $F_b = 1 - \sigma_f$, is chosen. There are 96, 96, and 160 grid intervals in the x , y , and σ_f directions, respectively. The horizontal resolution (Δx and Δy) is 10 m and the vertical resolution of σ_f is 1/160 ($\Delta z = 10$ m). The surface pressure is 1000 hPa, and the potential temperature is 303 K (isentropic). Four lateral boundaries of the Exner function perturbation are set to zero. The initial bubble has a constant potential temperature perturbation of 0.5 K within a radius of 50 m and then exponentially decaying to the boundaries. The formula of the thermal bubble is

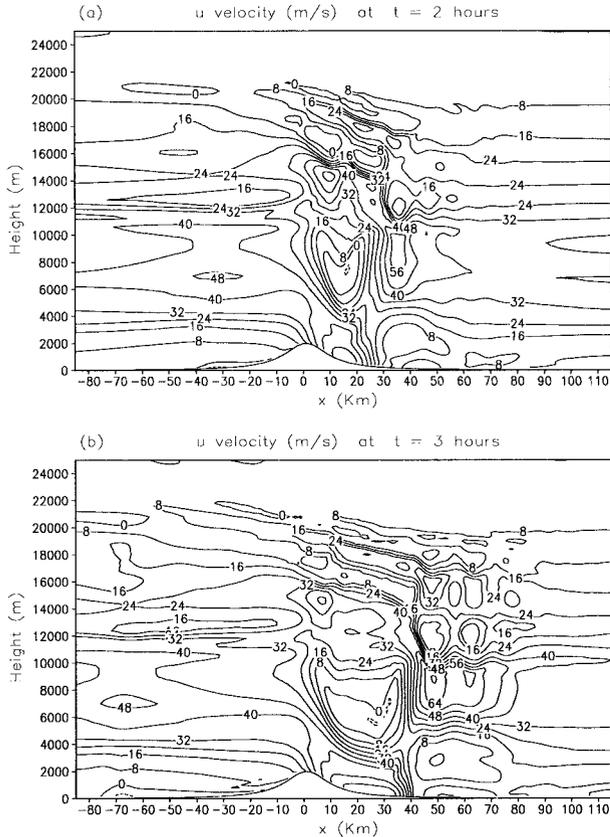


FIG. 8. The vertical cross sections of zonal wind (m s^{-1}) at (a) $t = 2$ h and (b) $t = 3$ h with the point-by-point relaxation and 0.1 tolerance number.

$$\theta' = \begin{cases} 0.5 & \text{if } r \leq 50 \text{ m,} \\ 0.5 \exp[(r - 50)^2/S^2] & \text{if } r > 50 \text{ m,} \end{cases}$$

where $r = \sqrt{(x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2}$ and $S = 100$ m. Here, x_0 and y_0 are the middle points of the domain in the x and y directions, respectively, and $z_0 = 260$ m. The time interval is 1 s, and the total integration time is 18 min. A point-by-point relaxation, V(1,1) cycle, and 0.1 tolerance number in the multigrid solver are chosen.

Figure 10 shows the three-dimensional potential temperature perturbation at 9, 12, 15, and 18 min. The thermal bubble moves upward after release because of buoyancy, and then it grows like a mushroom. The upper part of the bubble gets thinner and thinner as time increases and the bubble top breaks at 15 min. After 18 min of simulation, part of the bubble has moved out of the domain through the upper boundary. The average iteration number of the V(1,1) cycle is 1.07.

5. Summary

Here, we present a three-dimensional, fully compressible, nonhydrostatic model. To overcome the ex-

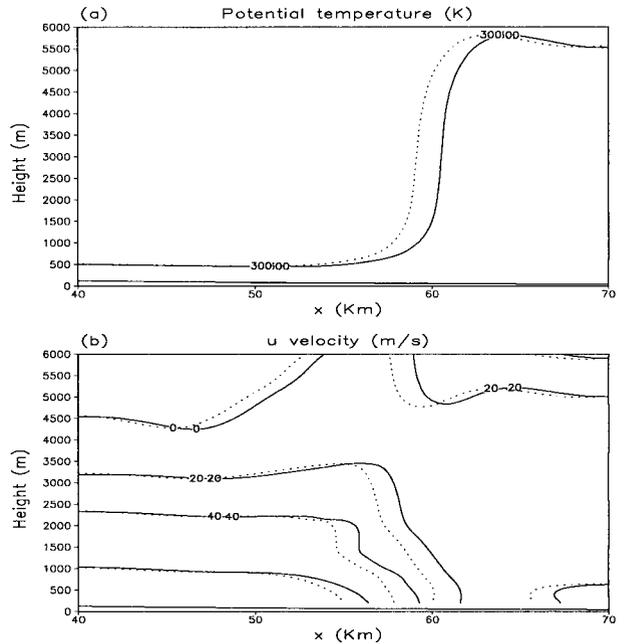


FIG. 9. (a) The potential temperature, and (b) the zonal wind with a hyperbolic tangent deviation function (solid lines) and a linear tangent deviation function (dashed lines) after 4-h integration in the wind-storm experiment. The formula of the hyperbolic tangent deviation function is the same as the one used in Fig. 3 except $C_1 = -3$ and $C_2 = -1$. Only a small domain on the lee side of the mountain is plotted.

ceedingly small time step required for high-frequency waves, a semi-implicit scheme is applied to this model. The use of the implicit scheme relaxes the extremely small time step requirement, but a linear system with a huge multiband sparse matrix of elliptic partial differential equations is encountered. The EPDEs become more difficult to solve when the cross-derivative terms are consistently integrated implicitly. The multigrid solver, mud3cr, is applied to solve this set of EPDEs.

The uniform grid spacing is required in the multigrid solver. To address this requirement, a new terrain-following coordinate, the flexible hybrid coordinate, is designed. Two functions are introduced in the FHC (σ_f): base (F_b) and deviation (F_d) functions. The base function is simply used to meet the uniform grid spacing in the multigrid solver, and the deviation function can be applied to adjust the vertical resolution, especially in the valley regions or stratosphere. Different combinations of base and deviation functions can present a variety of vertical structures of the σ_f surfaces. The recommended functions are exponential, hyperbolic tangent, and polynomial.

Three experiments are provided to demonstrate the performance of this nonhydrostatic model and the efficiency of the multigrid solver: one with linear mountain waves (Queney 1948), one with a downslope wind-storm (Lilly 1978), and one with a three-dimensional thermal bubble. In the nonhydrostatic linear mountain

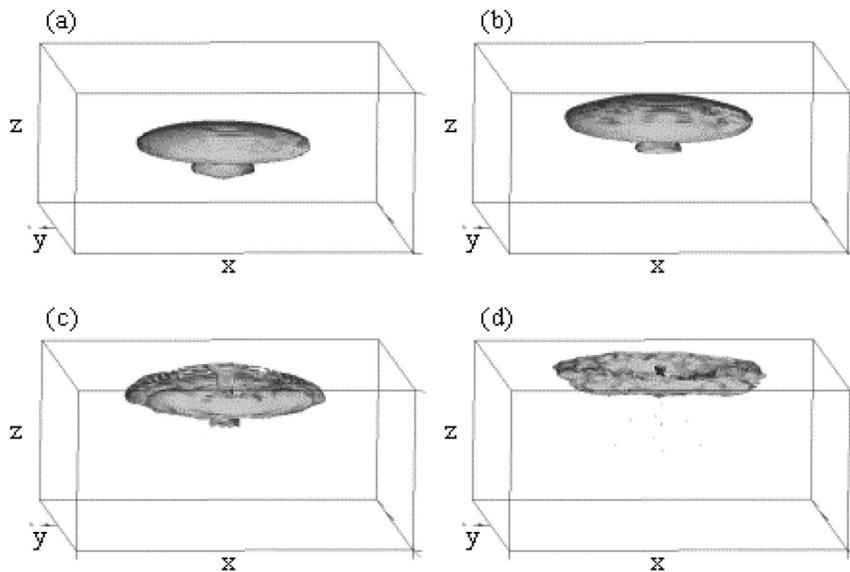


FIG. 10. The time evolution of the three-dimensional thermal bubble. The isotropic potential temperature perturbation of 0.07 K is plotted at (a) 9, (b) 12, (c) 15, and (d) 18 min. The domain size is 980, 980, and 1600 m in the x , y , and z directions, respectively.

wave case, the model results are in very good agreement with the analytical solution for all runs. A better surface pressure is obtained with a higher resolution near the surface, and a better potential temperature field in the lower troposphere is simulated with double grid intervals in each direction. This case is further used to diagnose the uncentered coefficient, α , and the value of 0.65 is considered as the optimal choice of this test in terms of the accuracy and efficiency. The surface pressure perturbation is calculated by solving the elliptic partial differential equations, and its solution is rather sensitive to the uncentered coefficient, more so than the potential temperature. The hydrostatic mountain wave simulations also show quite reasonable results. In the downslope windstorm experiment, the model reproduces the observed hydraulic jump and the magnitude of the downslope wind reasonably well. The simulated maximum wind occurs near the surface due to the implementation of an inviscid lower boundary condition. It is worth mentioning that the windstorm simulation results are very similar to those from the National Taiwan University–Purdue model in Doyle et al. (2000), which used a time-split explicit scheme to calculate high-frequency waves. The windstorm case is also used to examine the potential influence of the deviation function and the results do show some impact of the use of different deviation functions. A three-dimensional thermal bubble is tested to further demonstrate the reasonable performance and the robustness of this model.

The multigrid method is not as popular in the atmospheric sciences as it is in engineering or some other fields, since the coefficients of the EPDEs are quite complicated in terrain-following coordinates, as well as because the method is more sophisticated to code than

others. One of the major purposes of this study is to address the performance and efficiency of the multigrid method when applied to nonhydrostatic atmospheric models. Several tests with different tolerance numbers, different relaxation choices (point by point or line by line), and different sweeps of prerelaxation and postrelaxation are examined. The use of the 0.001 tolerance number does not improve the solutions, but requires more CPU time. Both nonhydrostatic and hydrostatic linear mountain waves, as well as the three-dimensional thermal bubble simulation, take one average $V(1,1)$ cycle with point-by-point relaxation to converge. It is noted that the very nonlinear problems, such as the windstorm case in this study, might need more iteration time to converge. Therefore, besides the tolerance number, the convergence speed may also depend on the property of the fluid, namely that convergence might be more difficult for a fast-changing fluid. Though the windstorm simulation requires two to three average $V(1,1)$ cycles with point-by-point relaxation or one to two average $V(1,1)$ cycles with line-by-line relaxation to converge, it is still quite efficient. Given enough accuracy, we therefore, consider that the point-by-point, $V(1,1)$ cycle relaxation with 0.1 tolerance number gives a very good performance for all experiments examined in this study on scalar machines, even when the grid aspect ratio ($\Delta z/\Delta x$) is much less than 1. It is worth mentioning that the line-by-line relaxation is superior to point-by-point relaxation for the windstorm case, which has a very nonlinear fluid and the grid aspect ratio is much less than 1, due to the order of the field indices of the model and the multigrid solver. The trade-off of using implicit methods is that extra memory is required for saving coefficients of the EPDEs.

Though we believe that the multigrid method is one of the best iteration methods for solving elliptic-type problems, more investigation is required. It seems that the conjugate residual method with a preconditioner also performs quite well, as discussed by Skamarock et al. (1997). A detailed comparison between these two methods will be of interest and we leave it for future work.

Acknowledgments. The authors would like to thank Dr. Jiun-dar Chern for his generous assistance and Dr. J. Adams at NCAR for his helpful discussion. The authors would also like to thank Dr. J. Bresch and two anonymous reviewers for their useful suggestions concerning this paper. NCAR is acknowledged for providing the multigrid software. This work is supported by NSF under Grant ATMS-9631572.

APPENDIX A

The Algorithm of a Three-Level Multigrid V-Cycle Relaxation

In Fig. 2, assume that we are solving a set of linear equations, $\mathbf{AU} = \mathbf{F}$, where \mathbf{A} is a $(n \times n)$ matrix and \mathbf{U} and \mathbf{F} are $(n \times 1)$ vectors. For any vector or matrix \mathbf{M} , \mathbf{M}^h indicates \mathbf{M} at the finest grid level, \mathbf{M}^{2h} indicates \mathbf{M} at the second finest grid level, etc. Here, we simply give five steps to illustrate the three-level $V(N_1, N_2)$ cycle relaxation scheme.

Step 1:

(a) Relax $\mathbf{A}^h\mathbf{U}^h = \mathbf{F}^h$ (prerelaxation). Assign an initial guess to \mathbf{U}^h at the finest grids. Use a relaxation scheme to estimate an approximation solution of \mathbf{U}^h with N_1 sweeps. The relaxation scheme could be a Jacobi or Gauss-Seidel type.

(b) Calculate $\mathbf{r}^h (= \mathbf{F}^h - \mathbf{A}^h\mathbf{U}^h = \mathbf{A}^h\mathbf{e}^h)$, which is a residual vector.

Step 2:

(a) Calculate $\mathbf{F}^{2h} (= \mathbf{I}_h^{2h}\mathbf{r}^h)$. Here, \mathbf{I}_h^{2h} is defined as a mapping matrix, which maps \mathbf{r}^h at finer grids (subscript) into \mathbf{F}^{2h} at next coarser grids (superscript). Similar definitions with different subscripts and superscripts are used later.

(b) Relax $\mathbf{A}^{2h}\mathbf{e}^{2h} = \mathbf{F}^{2h}$ (prerelaxation). Assign an initial guess 0 to \mathbf{e}^{2h} . Apply the relaxation scheme to solve the residual equation with N_1 sweeps and get an approximation solution of \mathbf{e}^{2h} .

(c) Calculate $\mathbf{r}^{2h} (= \mathbf{F}^{2h} - \mathbf{A}^{2h}\mathbf{e}^{2h})$.

Step 3:

(a) Calculate $\mathbf{F}^{4h} (= \mathbf{I}_{2h}^{4h}\mathbf{r}^{2h})$.

(b) Solve $\mathbf{A}^{4h}\mathbf{e}^{4h} = \mathbf{F}^{4h}$ to get the solution \mathbf{e}^{4h} at the coarsest grids using direct or iterative methods. The number of the coarsest grid intervals is not required to be a power of 2.

Step 4:

(a) Update $\mathbf{e}^{2h} (\leftarrow \mathbf{e}^{2h} + \mathbf{I}_{4h}^{2h}\mathbf{e}^{4h})$ (coarse grid correction). Here, \mathbf{I}_{4h}^{2h} is a mapping matrix, which maps \mathbf{e}^{4h} at coarser grids (subscript) into finer grids (superscript).

(b) Relax $\mathbf{A}^{2h}\mathbf{e}^{2h} = \mathbf{F}^{2h}$ (postrelaxation) for N_2 sweeps with the new initial guess \mathbf{e}^{2h} .

Step 5:

(a) Update $\mathbf{U}^h (\leftarrow \mathbf{U}^h + \mathbf{I}_{2h}^h\mathbf{e}^{2h})$ (coarse grid correction).

(b) Relax $\mathbf{AU}^h = \mathbf{F}^h$ (postrelaxation) for N_2 sweeps with the new initial guess \mathbf{U}^h .

Repeat steps 1–5 until $r_{\max} < \text{tol}$ or until the maximum outer iteration number of V cycle is reached. Here, r_{\max} is an error measurement, which is defined in section 3, and tol is a tolerance number.

The operator moving a vector from finer grids to coarser grids is called the restriction operator, such as \mathbf{I}_h^h and \mathbf{I}_{2h}^h . The simplest restriction operator is injection, in which the vector on coarser grids simply takes its values directly from the corresponding finer grid points. Another popular one is the full weighting operator (Briggs 1987; Briggs et al. 2000). In contrast, the operator moving a vector from coarser grids to finer grids is called the interpolation or prolongation operator, such as \mathbf{I}_h^h and \mathbf{I}_{4h}^{2h} . In general, a linear interpolation is good enough for the multigrid method.

APPENDIX B

The Coefficients of the Elliptic Partial Differential Equation

Here are the formulas for the left-hand-side coefficients and source/sink term r of Eq. (18):

$$\begin{aligned}
 C_{xx} &= C_{yy} = Am_p^2 \bar{\theta} \bar{\pi}, & C_{zz} &= C_{xx} [B_x^2 + B_y^2 + E^{-1} B_z^2 / m_p^2], & C_{xz} &= -2C_{xx} B_x, & C_{yz} &= -2C_{yy} B_y, \\
 C_x &= Am_p^2 \bar{\pi} \left(\frac{\partial \bar{\theta}}{\partial x} - B_x \frac{\partial \bar{\theta}}{\partial \sigma_f} \right), & C_y &= Am_p^2 \bar{\pi} \left(\frac{\partial \bar{\theta}}{\partial y} - B_y \frac{\partial \bar{\theta}}{\partial \sigma_f} \right), \\
 C_z &= -A \bar{\pi} \left[m_p^2 \left(\frac{\partial (\bar{\theta} B_x)}{\partial x} - B_x \frac{\partial (\bar{\theta} B_x)}{\partial \sigma_f} + \frac{\partial (\bar{\theta} B_y)}{\partial y} - B_y \frac{\partial (\bar{\theta} B_y)}{\partial \sigma_f} \right) - B_z E^{-1} \frac{\partial (\bar{\theta} B_z)}{\partial \sigma} + \bar{\theta} B_z^2 E^{-2} \frac{\partial E}{\partial \sigma_f} \right], & C_e &= -1, \\
 r &= -\frac{(1 - \alpha)}{\alpha} \left(C_{xx} \frac{\partial^2 \pi'^n}{\partial x^2} + C_{yy} \frac{\partial^2 \pi'^n}{\partial y^2} + C_{zz} \frac{\partial^2 \pi'^n}{\partial \sigma_f^2} + C_{xz} \frac{\partial^2 \pi'^n}{\partial x \partial \sigma_f} + C_{yz} \frac{\partial^2 \pi'^n}{\partial y \partial \sigma_f} + C_x \frac{\partial \pi'^n}{\partial x} + C_y \frac{\partial \pi'^n}{\partial y} + C_z \frac{\partial \pi'^n}{\partial \sigma_f} \right) \\
 &\quad + C_e \pi'^* + B \alpha B_z E^{-1} \left[\Delta t g \frac{\partial}{\partial \sigma_f} \left(\frac{\theta'^*}{\theta} \right) - \alpha \Delta t^2 (1 - \alpha) g \frac{\partial}{\partial \sigma_f} \left(\frac{w^*}{\theta} \frac{\partial \bar{\theta}}{\partial \sigma_f} B_z \right) \right]
 \end{aligned}$$

$$\begin{aligned}
& - B\alpha B_z E^{-2} \frac{\partial E}{\partial \sigma_f} \left[\Delta t g \frac{\theta'^*}{\theta} + \left(1 - \alpha \Delta t^2 \frac{(1-\alpha)}{\theta} g B_z \frac{\partial \bar{\theta}}{\partial \sigma_f} \right) w^* \right] \\
& + B \left[m_p^2 \left(\frac{\partial u^*/m_u}{\partial x} - B_x \frac{\partial u^*/m_u}{\partial \sigma_f} + \frac{\partial v^*/m_v}{\partial y} - B_y \frac{\partial v^*/m_v}{\partial \sigma_f} \right) + B_z (1 - \alpha + \alpha E^{-1}) \frac{\partial w^*}{\partial \sigma_f} \right],
\end{aligned}$$

where

$$A = (R/C_v) \Delta t^2 C_p \alpha^2, \quad \text{and} \quad B = (R/C_v) \Delta t \bar{\pi}.$$

REFERENCES

- Adams, J. C., 1989: MUDPACK: Multigrid portable FORTRAN software for the efficient solution of linear elliptic partial differential equations. *Appl. Math. Comput.*, **34**, 113–146.
- , 1991: Recent enhancements in MUDPACK, a multigrid software package for elliptic partial differential equations. *Appl. Math. Comput.*, **43**, 79–93.
- , and P. Smolarkiewicz, 2001: Modified multigrid for 3D elliptic equations with cross-derivatives. *Appl. Math. Comput.*, **121**, 301–312.
- , R. Garcia, B. Gross, J. Hack, D. Haidvogel, and V. Pizzo, 1992: Applications of multigrid software in the atmospheric sciences. *Mon. Wea. Rev.*, **120**, 1447–1458.
- Arakawa, A., and V. R. Lamb, 1977: Computational design of the basic dynamical processes of the UCLA general circulation model. *Methods in Computational Physics*, J. Chang Ed., Vol. 17, Academic Press, 173–265.
- Bates, J. R., F. H. M. Semazzi, R. W. Higgins, and S. R. M. Barros, 1990: Integration of the shallow water equations on the sphere using a vector semi-Lagrangian scheme with a multigrid solver. *Mon. Wea. Rev.*, **118**, 1615–1627.
- Bowman, K. P., and J. Huang, 1991: A multigrid solver for the Helmholtz equation on a semiregular grid on the sphere. *Mon. Wea. Rev.*, **119**, 769–775.
- Brandt, A., 1977: Multi-level adaptive solutions to boundary-value problems. *Math. Comput.*, **31**, 333–390.
- Briggs, W. L., 1987: *A Multigrid Tutorial*. Society for Industrial and Applied Mathematics, 90 pp.
- , V. E. Henson, and S. F. McCormick, 2000: *A Multigrid Tutorial*. Society for Industrial and Applied Mathematics, 193 pp.
- Chen, S. H., 1999: Part I: The development of a nonhydrostatic model. Ph.D. thesis, Purdue University, 83 pp.
- Ciesielski, P. E., S. R. Fulton, and W. H. Schubert, 1986: Multigrid solution of an elliptic boundary value problem from tropical cyclone theory. *Mon. Wea. Rev.*, **114**, 797–807.
- Crowley, W. P., 1968: Numerical advection experiments. *Mon. Wea. Rev.*, **96**, 1–11.
- Cullen, M. J. P., 1990: A test of a semi-implicit integration technique for a fully compressible non-hydrostatic model. *Quart. J. Roy. Meteor. Soc.*, **116**, 1253–1258.
- Doyle, J. D., and Coauthors, 2000: An intercomparison of model-predicted wave breaking for the 11 January 1972 Boulder windstorm. *Mon. Wea. Rev.*, **128**, 901–914.
- Durran, D. R., and J. B. Klemp, 1983: A compressible model for the simulation of moist mountain waves. *Mon. Wea. Rev.*, **111**, 2341–2361.
- Engquist, B., and E. Luo, 1997: Convergence of a multigrid method for elliptic equations with highly oscillatory coefficients. *SIAM J. Numer. Anal.*, **34**, 2254–2273.
- Fulton, S. R., P. E. Ciesielski, and W. H. Schubert, 1986: Multigrid methods for elliptic problems: A review. *Mon. Wea. Rev.*, **114**, 943–959.
- Gadd, A. J., 1978: A split explicit integration scheme for numerical weather prediction. *Quart. J. Roy. Meteor. Soc.*, **104**, 569–582.
- Golding, B. W., 1992: An efficient non-hydrostatic forecast model. *Meteor. Atmos. Phys.*, **50**, 89–103.
- Hiptmair, R., 1998: Multigrid method for Maxwell's equations. *SIAM J. Numer. Anal.*, **36**, 204–225.
- Hsu, W.-R., and W.-Y. Sun, 2001: A time-split, forward-backward numerical model for solving a nonhydrostatic and compressible system of equations. *Tellus*, **53A**, 279–299.
- Ikawa, M., 1988: Comparison of some schemes for nonhydrostatic models with orography. *J. Meteor. Soc. Japan*, **66**, 753–776.
- Juang, H.-M. H., 1992: A spectral fully compressible nonhydrostatic mesoscale model in hydrostatic sigma coordinates: Formulation and preliminary results. *Meteor. Atmos. Phys.*, **50**, 75–88.
- , 2000: The NCEP mesoscale spectral model: A revised version of the nonhydrostatic regional spectral model. *Mon. Wea. Rev.*, **128**, 2329–2362.
- Kapitza, H., and D. P. Eppel, 1992: The non-hydrostatic mesoscale model GESIMA. Part I: Dynamical equations and tests. *Beitr. Phys. Atmos.*, **65**, 129–145.
- Klemp, J. B., and R. B. Wilhelmson, 1978: The simulation of three-dimensional convective storm dynamics. *J. Atmos. Sci.*, **35**, 1070–1096.
- Laprise, R., 1992: The Euler equations of motion with hydrostatic pressure as an independent variable. *Mon. Wea. Rev.*, **120**, 197–207.
- Lilly, D. K., 1978: A severe downslope windstorm and aircraft turbulence event induced by a mountain wave. *J. Atmos. Sci.*, **35**, 59–77.
- Mendez-Nunez, L. R., and J. J. Carroll, 1994: Application of the MacCormack scheme to atmospheric nonhydrostatic models. *Mon. Wea. Rev.*, **122**, 984–1000.
- Mertens, R., H. de Gersem, R. Belmans, K. Hameyer, D. Lahaye, S. Vandewalle, and D. Roose, 1998: An algebraic multigrid method for solving very large electromagnetic systems. *IEEE Trans. Magn.*, **34**, 3327–3330.
- Miller, P. P., and D. R. Durran, 1991: On the sensitivity of downslope windstorms to the asymmetry of the mountain profile. *J. Atmos. Sci.*, **48**, 1457–1473.
- Pinty, J.-P., R. Benoit, E. Richard, and R. Laprise, 1995: Simple tests of a semi-implicit semi-Lagrangian model on 2D mountain wave problems. *Mon. Wea. Rev.*, **123**, 3042–3058.
- Queney, P., 1948: The problem of airflow over mountains: A summary of theoretical studies. *Bull. Amer. Meteor. Soc.*, **29**, 16–26.
- Richard, E., P. Mascart, and E. C. Nickerson, 1990: Examples of the role of surface friction in downslope windstorms. *Meteor. Atmos. Phys.*, **43**, 163–172.
- Robert, A., 1969: The integration of a spectral model of the atmosphere by the implicit method. *Proc. WMO-IUGG Symp. on NWP*, Vol. VII, Tokyo, Japan, Japan Meteorological Agency, 19–24.
- , 1993: Bubble convection experiments with a semi-implicit formulation of the Euler equations. *J. Atmos. Sci.*, **50**, 1865–1873.
- Simmons, A. J., and D. M. Burridge, 1981: An energy and angular-momentum conserving vertical finite-difference scheme and hybrid vertical coordinates. *Mon. Wea. Rev.*, **109**, 758–766.
- , and R. Strüfing, 1981: An energy and angular momentum conserving finite-difference scheme, hybrid coordinates and medium-range weather prediction. ECMWF Tech. Rep. 28, 67 pp.
- , and C. Temperton, 1996: Stability of a two-time-level semi-

- implicit integration scheme for gravity-wave motion. ECMWF Tech. Memo. 226, 33 pp.
- Skamarock, W. C., P. K. Smolarkiewicz, and J. B. Klemp, 1997: Preconditioned conjugate-residual solvers for Helmholtz equations in nonhydrostatic models. *Mon. Wea. Rev.*, **125**, 587–599.
- Smith, R. B., 1985: On severe downslope winds. *J. Atmos. Sci.*, **42**, 2597–2603.
- Sun, W.-Y., 1993: Numerical experiments for advection equation. *J. Comput. Phys.*, **108**, 264–271.
- Tanguay, M., A. Robert, and R. Laprise, 1990: A semi-implicit semi-Lagrangian fully compressible regional forecast model. *Mon. Wea. Rev.*, **118**, 1970–1980.
- Tapp, M. C., and P. W. White, 1976: A non-hydrostatic mesoscale model. *Quart. J. Roy. Meteor. Soc.*, **102**, 277–296.
- Thomas, S., C. Girard, R. Benoit, M. Desgagne, and P. Pellerin, 1998: A new adiabatic kernel for the MC2 model. *Atmos.–Ocean*, **36**, 241–270.
- , ——, G. Doms, and U. Schattler, 2000: Semi-implicit scheme for the DWD Lokal-Modell. *Meteor. Atmos. Phys.*, **73**, 105–125.