A Control-Volume Model of the Compressible Euler Equations with a Vertical Lagrangian Coordinate

XI CHEN AND NATALIA ANDRONOVA

Atmospheric Oceanic and Space Science, University of Michigan, Ann Arbor, Ann Arbor, Michigan

BRAM VAN LEER

Aerospace Engineering, University of Michigan, Ann Arbor, Ann Arbor, Michigan

JOYCE E. PENNER, JOHN P. BOYD, AND CHRISTIANE JABLONOWSKI

Atmospheric Oceanic and Space Science, University of Michigan, Ann Arbor, Ann Arbor, Michigan

SHIAN-JIANN LIN

NOAA/Geophysical Fluid Dynamics Laboratory, Princeton, New Jersey

(Manuscript received 17 April 2012, in final form 23 December 2012)

ABSTRACT

Accurate and stable numerical discretization of the equations for the nonhydrostatic atmosphere is required, for example, to resolve interactions between clouds and aerosols in the atmosphere. Here the authors present a modification of the hydrostatic control-volume approach for solving the nonhydrostatic Euler equations with a Lagrangian vertical coordinate. A scheme with low numerical diffusion is achieved by introducing a low Mach number approximate Riemann solver (LMARS) for atmospheric flows. LMARS is a flexible way to ensure stability for finite-volume numerical schemes in both Eulerian and vertical Lagrangian configurations. This new approach is validated on test cases using a 2D (x–z) configuration.

1. Introduction

Most of the current global climate models are based on equations that assume hydrostatic equilibrium. These models resolve processes whose horizontal scale is significantly larger than the vertical scale. However, if a global model aims to resolve motions whose horizontal and vertical scales are similar, the model must also include nonhydrostatic effects (Daley 1988). One difficulty with the nonhydrostatic equations is that the fast sound waves, generated by the model's equations, can travel in all directions, vertically and horizontally, and thus, require special computational approaches (e.g., implicit methods; Skamarock and Klemp 1992) and/or small time steps in explicit time stepping schemes. Therefore, the main question is how to formulate an efficient numerical scheme for small-scale nonhydrostatic models, which has the ability to correctly and stably represent the important small atmospheric interactions at the model's resolution limits (Smolarkiewicz et al. 2001; Skamarock and Klemp 2008).

A large number of hydrostatic and nonhydrostatic models use pressure or pressure-based terrain-following sigma or hybrid coordinates as the vertical coordinate (Phillips 1956; Smagorinsky 1963; Kasahara 1974; Bates et al. 1993; Miller and Pearce 1974; Miller and White 1984; Xue and Thorpe 1991; Juang 1992; Skamarock and Klemp 2008). Laprise (1992) suggested that hydrostatic-pressure coordinates could be used advantageously in nonhydrostatic atmospheric models. Since the mass in the layers between adjacent coordinate surfaces is proportional to the increment in the vertical coordinate across the layer, this coordinate is often referred to as a mass coordinate. A ''floating'' mass coordinate, which does not allow mass to flow across vertical layers, is

Corresponding author address: Xi Chen, Atmospheric Oceanic and Space Science, University of Michigan, 2455 Hayward St., Ann Arbor, MI 48109-2143.
E-mail: xic@umich.edu

DOI: 10.1175/MWR-D-12-00129.1

called a vertical Lagrangian coordinate. A few hydrostatic climate models have been developed using this formulation, including the National Center for Atmospheric Research (NCAR) Community Atmosphere Model (CAM) versions 4 and 5 (Neale et al. 2010) and the National Oceanic and Atmospheric Administration (NOAA) Geophysical Fluid Dynamics Laboratory (GFDL) models (Putman and Lin 2008; Donner et al. 2011) based on the finite-volume dynamical core by Lin (2004). A major advantage of applying a vertical Lagrangian coordinate is that the 3D motion can be reformulated into pure horizontal 2D flow within the floating Lagrangian layers, with the resulting system closely resembling that of the shallow-water system (Lin 2004). Developing a nonhydrostatic model based on a vertical Lagrangian coordinate could allow the dynamics component of general circulation models (GCMs) to switch between different representations with hydrostatic dynamics in some regions and nonhydrostatic dynamics in other regions where higher resolution is desired. Klemp et al. (2007) implemented a vertical mass coordinate in a nonhydrostatic model. However, their vertical mass coordinate was not implemented using a Lagrangian formulation. The GFDL Finite-Volume Cubed Sphere (GFDL FVcubed) model (Donner et al. 2011), which is based on a vertical Lagrangian coordinate, has a nonhydrostatic option. However, an explicit divergence damping term was required to maintain its stability. Numerical damping in atmospheric modeling in general, whether accomplished implicitly through the numerical scheme or explicitly through the addition of specific terms to the equations, should only be large enough to maintain a smooth and stable integration.

In this paper we present the development of a 2D (x–z) nonhydrostatic dynamical model based on the use of a generalized Lagrangian vertical coordinate, which was also adopted in the well-known Lin–Rood hydrostatic dynamical core (Lin and Rood 1996, 1997; Lin 2004). The multidimensional Flux-Form Semi-Lagrangian (FFSL) Lin–Rood dynamical core simulates the conservative, monotonic advection of the prognostic variables, and uses a floating vertically Lagrangian finite-volume (FV) representation of the model equations with a conservative remapping algorithm in the vertical direction. The Lagrangian coordinate requires periodic remapping to a reference grid in order to avoid severe deformation of the vertical mesh, which would occur, for example, if layers with overlapping interfaces develop. The horizontal numerical algorithm of the Lin–Rood dynamical core is based on a staggered C–D grid approach (Arakawa and Lamb 1977). This FFSL FV algorithm has been adopted in several atmospheric GCMs (e.g., CAM and GFDL).

The development of a global nonhydrostatic climate model represents a computational challenge. However, it has become feasible to embed nonhydrostatic regions within a hydrostatic model (Yeh et al. 2002). While an FV model formulation based on the flux form of the equations is favorable for maintaining mass and momentum conservation when merging the hydrostatic and nonhydrostatic regions, grid staggering in a C–D fashion for the nonhydrostatic atmosphere may limit the ability of the model to perform in a stable and accurate fashion (Skamarock 2008; Ullrich et al. 2010; Whitehead et al. 2011).

Here, we explore the use of a vertically Lagrangian nonhydrostatic model. The purpose of this exploration is to ultimately be able to join hydrostatic and nonhydrostatic formulations in an adaptive model as grid resolution is decreased. Because a large number of general circulation models currently use a Lagrangian hydrostatic formulation, and our previous work explored adaptive grid techniques in a Lagrangian hydrostatic model (Jablonowski et al. 2006), it is more straightforward and easy to do this using the Lagrangian formulation in both models. This will hopefully lead to the ability to seamlessly treat both hydrostatic and nonhydrostatic regimes and allow adaptive mesh refinement using this framework. Here, we develop a method for solving the 2D (x–z) nonhydrostatic equations in Cartesian geometry using a finite-volume approach based on an unstaggered grid together with a generalized Lagrangian vertical coordinate. We do not filter acoustic waves in order to minimize any added numerical diffusion caused by filtering. Since the nonhydrostatic equations are nonlinear, no analytical solution can be used to validate the accuracy of our results. Thus, we also applied our scheme using an Eulerian coordinate system to act as a reference for comparison purposes. We test the method based on both an Eulerian and a Lagrangian formulation, using the 2D warm bubble tests of Robert (1993) and propagating gravity waves.

Several advantages of the current scheme are as follows:

1) No divergence damping is needed.
2) The use of an unstaggered grid simplifies the numerical representation of the advection equations.
3) We developed a fast method for evaluating the fluxes using a new approximate Riemann solver for low speed flow.
4) A Lagrangian vertical coordinate was introduced to facilitate the switching between a Lagrangian hydrostatic and nonhydrostatic treatment, since the variables in both treatments share the same definition.

This approach allows us to easily join the efficient Lagrangian hydrostatic approach with the nonhydrostatic approach in the GCM.

Some advantages inherited from the finite-volume framework as are follows:

5) Its built-in physical conservation laws.
6) It is free of Gibbs oscillations.

One disadvantage of our current scheme is that we implement a vertically explicit scheme. In weather or climate models where the ratio of horizontal to vertical grid spacing might be of order 10 to 100, it might be more efficient to implement a vertically implicit scheme in order to allow for longer time steps. However, in this work, in order to examine the numerical properties of our nonstaggered, Riemann solver-based algorithm, we kept the numerical treatments simple and applied them to a problem with similar vertical and horizontal grid spacing. We also do not use a limiter for the reconstruction of the conservative variables profiles in order to avoid the added diffusion when limiters are adopted. This approach represents a first step toward a more generally applicable formulation.

The paper is organized as follows: in section 2 we present a 2D (x–z) version of the fully compressible Euler equations. Section 3 introduces the numerical technique for their solution. In section 4 the results of the model tests are discussed. We have conducted some quantitative analysis and discussions in section 5. Section 6 presents the conclusions.

2. Model equations

To be able to embed our nonhydrostatic model into a hydrostatic model with a vertical Lagrangian coordinate, we focus on developing the nonhydrostatic model with a similar vertical coordinate. However, the Eulerian vertical coordinate can also be used based on a natural extension of the technique we introduce. Thus, we also include some results using the nonhydrostatic model with an Eulerian coordinate configuration. We introduce the control equations using the vertical Lagrangian coordinate in section 2a, and the equations using the Eulerian coordinate in section 2b.

a. The finite-volume equations in a vertical Lagrangian coordinate

1) THE NONHYDROSTATIC FORMULATION

The model equations are the fully compressible 2D (x, z) Euler conservation equations in flux form with a vertical Lagrangian coordinate. Because of the latter, the model layers are material impenetrable surfaces and the bottom surface is terrain following. This eliminates the need for the vertical advection terms and renders the equations one-dimensional. Instead, vertical transport is represented by the remapping mechanism.

The mass conservation law is written in the following form:

$$\frac{\partial \pi}{\partial t} + \frac{\partial \pi u}{\partial x} = 0, \tag{1}$$

where π is interpreted as a pseudodensity, which is the density multiplied by the vertical geopotential gradient within the Lagrangian FV, and has the units of pressure. Here u is the magnitude of the horizontal wind. Here π is defined as

$$\pi = \frac{\partial p^*}{\partial s} = -\rho g \frac{\partial z}{\partial s} = -\rho \frac{\partial \Phi}{\partial s}, \tag{2}$$

where ρ, g, Φ, and p^* are the nonhydrostatic density, gravity, geopotential, and hydrostatic pressure, respectively, and s is a generalized vertical coordinate which is the integer index of each of the layer interfaces, numbered in the top-down direction. Thus, $\partial/\partial s$ denotes the difference of the value of any parameter between two Lagrangian layer interfaces.

The horizontal momentum equation is

$$\frac{\partial \pi u}{\partial t} + \frac{\partial}{\partial x}(\pi u u + \Psi) = -\frac{\partial}{\partial s}\left(p \frac{\partial \Phi}{\partial x}\right), \tag{3}$$

where Ψ is defined as the nonhydrostatic pressure p multiplied by the vertical geopotential gradient:

$$\Psi = -p \frac{\partial \Phi}{\partial s}. \tag{4}$$

Since π is interpreted as a pseudodensity, the left-hand side of Eq. (3) is consistent with the general 1D flux form momentum equation.

The vertical momentum equation is

$$\frac{\partial \pi w}{\partial t} + \frac{\partial \pi w u}{\partial x} = g \frac{\partial p'}{\partial s}, \tag{5}$$

where w is the vertical velocity, $p' = p - p^*$ is the deviation from the hydrostatic pressure, and $\partial p'/\partial s = \partial p/\partial s - \pi$ is the perturbation of the pressure from hydrostatic balance between layers.

The first law of thermodynamics provides the conservation of the potential temperature equation:

$$\frac{\partial \Theta}{\partial t} + \frac{\partial \Theta u}{\partial x} = 0, \tag{6}$$

where $\Theta = \pi\theta/p_0^{\kappa}$, is a scaled pseudopotential temperature density with θ the potential temperature, p_0 is a

constant reference pressure at the surface with $p_0 = 1000$ hPa, and κ is the ratio of the gas constant R_d and the heat capacity at constant pressure c_p for dry air.

In these equations, the variables π, πu, πw, and Θ are treated as prognostic variables. Two additional equations are needed to predict the full set of nonhydrostatic variables, and we add equations for the geopotential Φ and the nonhydrostatic pressure p. The geopotential advection equation comes from the definition of the vertical velocity in z coordinates:

$$\frac{\partial \Phi}{\partial t} + u\frac{\partial \Phi}{\partial x} = wg. \tag{7}$$

The equation for the nonhydrostatic pressure is derived from the equation of state:

$$p = \left(-\frac{R_d\Theta}{\partial\Phi/\partial s}\right)^{\gamma}, \tag{8}$$

where $\gamma = 1/(1 - \kappa)$ Since the index s is numbered in top-down direction, $\partial\Phi/\partial s$ is negative.

Equations (1), (3), and (5)–(8) define our nonhydrostatic system.

2) THE HYDROSTATIC FORMULATION

In the hydrostatic system the vertical velocity w is not treated as a prognostic variable. Our system of equations is fully consistent with a hydrostatic system if we add the assumption that $p' = 0$ (i.e., p is simply the hydrostatic pressure p^*). In addition, Eq. (5) is not used, and the vertical velocity may be diagnostically derived from Eq. (7). The pressure p^* at the layer interfaces is calculated by (for level index $k > 1$):

$$(p_I^*)_k = p_{k-1/2}^* = p_{\text{top}}^* + \sum_{n=1}^{n=k-1} \pi_n, \tag{9}$$

where $(p_I^*)_{k=1} = p_{k=1/2}^* = p_{\text{top}}^*$ is the pressure at the model top. The layer mean pressure p^* is calculated by

$$p^* = \left(\kappa\frac{\partial p_I^*/\partial s}{\partial p_I^{*\kappa}/\partial s}\right)^{\gamma} \tag{10}$$

and the equation of state [Eq. (8)] is modified to calculate the geopotential:

$$\frac{\partial \Phi}{\partial s} = -c_p\Theta\frac{\partial p_I^{*\kappa}/\partial s}{\partial p_I^*/\partial s}. \tag{11}$$

Equations (9), (10), and (11) are the auxiliary equations needed to calculate Ψ and $-\partial(p\partial\Phi/\partial x)/\partial s$ in Eq. (3). Thus, Eqs. (1), (3), (6), (9), (10), and (11) complete the hydrostatic system.

b. The finite-volume equations in an Eulerian coordinate system

The layers in the Eulerian coordinate are stationary, so that vertical fluxes across layer boundaries take place at the layer interfaces. The prognostic variables in the Eulerian coordinate are ρ, ρu, ρw, $\tilde{\Theta}$, with the scaled potential temperature density $\tilde{\Theta} = \rho\theta/p_0^{\kappa}$, and the corresponding set of equations are as follows:

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u}{\partial x} + \frac{\partial \rho w}{\partial z} = 0, \tag{12}$$

$$\frac{\partial \rho u}{\partial t} + \frac{\partial}{\partial x}(\rho uu + p) + \frac{\partial \rho uw}{\partial z} = 0, \tag{13}$$

$$\frac{\partial \rho w}{\partial t} + \frac{\partial \rho wu}{\partial x} + \frac{\partial}{\partial z}(\rho ww + p') = 0, \tag{14}$$

$$\frac{\partial \tilde{\Theta}}{\partial t} + \frac{\partial \tilde{\Theta} u}{\partial x} + \frac{\partial \tilde{\Theta} w}{\partial z} = 0, \tag{15}$$

$$p = (R_d\tilde{\Theta})^{\gamma}. \tag{16}$$

Note that the position of the layers is prescribed, so there is no equation for the geopotential.

3. Solution technique

In this section, we mainly discuss the discretization technique using the vertical Lagrangian coordinate configuration. The discretization using the Eulerian coordinate system is analogous.

Our dynamical core consists of six equations [Eqs. (1), (3), (5), (6), (7), and (8)] for six variables: $Q = (\pi, u, w, \Theta, p, \Phi)$. Since Φ is not a conservative variable, Eq. (7) is updated using the advective form. We use the flux form for the Eqs. (1), (3), (5), and (6), which has the general form:

$$\frac{\partial \mathbf{R}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} = \frac{\partial \mathbf{S}}{\partial s}, \tag{17}$$

where $\mathbf{R} = (\pi, \pi u, \pi w, \Theta)$ is a vector of the π-weighted variables, $\mathbf{F} = (\pi u, \pi uu + \Psi, \pi wu, \Theta u)$ is the flux vector, and $\mathbf{S} = (0, -p\partial\Phi/\partial x, gp', 0)$ is a source vector. The use of the equations in the flux form assures conservation of mass, momentum, and potential temperature.

In our numerical representation of the model equations, we use an unstaggered grid and place the variables π, πu, πw, Θ, p, and $\partial\Phi/\partial s$ at the center of the cell (the A grid) so that they represent the volume mean values of these variables. The volume mean values of u and w are calculated as: $u = (\pi u)/\pi$ and $w = (\pi w)/\pi$, respectively.

An upwind method for determining the flux is used. The flux vector $\mathbf{F}$ is placed at the FV horizontal interfaces and is split according to

$$\mathbf{F} = (\pi_b u_{1/2}, (\pi u)_b u_{1/2} + \Psi_{1/2}, (\pi w)_b u_{1/2}, \Theta_b u_{1/2}) = \mathbf{R}_b u_{1/2} + \Psi 1/2_{1/2}(0,1,0,0), \quad (18)$$

with $u_{1/2}$ as the interface velocity and the index b as an upwind indicator: $b = l$, if $u_{1/2} > 0$, and the scalar advection terms are chosen from the left (l) side of the interface; and $b = r$, if $u_{1/2} \leq 0$, and the scalar advection terms are chosen from the right (r) side of the interface. Lin (2004) used a similar form to transport moisture in a general circulation model, with the horizontal velocity defined on the interface of the FV using a C–D grid and achieved a second-order overall accuracy. However, in a nonhydrostatic model, the vertical velocity needs to be taken into account, and the expansion of the staggered grid approach to three dimensions in the treatment of vertical velocity is not intuitive. S. J. Lin (2007, unpublished manuscript) used an A-grid to treat the vertical velocity. However, the use of a staggered grid for the horizontal velocity and an unstaggered grid for the vertical velocity introduces an inconsistency.

Liou (2006) invented an Advection Upstream Splitting Method (AUSM$^+$-up) for accurate flux calculations for all flow speeds. This method was also based on an upwind method with the flux as presented in Eq. (18). Ullrich et al. (2010) applied this method to a shallow-water model and achieved third- and fourth-order accuracy.

However, to make the AUSM$^+$-up method suitable for all flow speeds (i.e., Mach number $M \sim 1$ and $M \gg 1$), the calculation of the interface flux vector is relatively complicated, and as a result, is computationally intensive. For most atmospheric phenomena, the Mach number is small ($M \ll 1$) and it is acceptable to assume the acoustic speed is locally constant. Thus, in order to achieve a computationally economic scheme, we invented the Low Mach number Approximate Riemann Solver (LMARS) to solve the system of equations [Eq. (17)] in the work described here. In the following sections, we show that the LMARS method has a simple form, which saves a substantial number of computational steps, but retains the accuracy of the AUSM$^+$-up method. Similar to the AUSM$^+$-up method, the LMARS consists of two steps. For the first step, the interface velocity $u_{1/2}$ and the pseudopressure $\Psi_{1/2}$ are calculated by solving a Riemann problem. Note that in the Eulerian coordinate the pseudopressure $\Psi_{1/2}$ is replaced by the real pressure $p_{1/2}$. For the second step, the fluxes are updated using Eq. (18). The $\mathbf{R}_b$ can be acquired by any kind of interpolation scheme. As an aside, the AUSM$^+$-up method is slightly different in that it does not calculate the interface velocity in the first step but calculates the interface mass flux instead. To achieve low diffusivity, we use a conservative five-point central polynomial interpolation scheme. Here we provide the expression of the interface values of η, where η could be velocity, pseudodensity, or pseudopressure, etc., at the cell i, where i could be either the horizontal or vertical index:

$$\eta_{i-1/2} = -\frac{1}{20}\eta_{i-2} + \frac{9}{20}\eta_{i-1} + \frac{47}{60}\eta_i - \frac{13}{60}\eta_{i+1} + \frac{1}{30}\eta_{i+2}, \quad (19)$$

$$\eta_{i+1/2} = \frac{1}{30}\eta_{i-2} - \frac{13}{60}\eta_{i-1} + \frac{47}{60}\eta_i + \frac{9}{20}\eta_{i+1} - \frac{1}{20}\eta_{i+2}. \quad (20)$$

On the horizontal boundaries, we have 4 ''ghost cells,'' which allow us to use the expression above for boundary cells; however, we do not use ghost cells for the vertical boundaries; thus, Eqs. (19) and (20) cannot be applied at the boundaries. Here we use the top boundary cells 1 and 2 as an example to describe the top boundary condition; the two cells at the bottom boundary can be treated analogously with reversing the index number.

We should point out that the interpolated profile of η by the polynomials is more stable when it is evaluated near the middle of the cells (i.e., the third cell's control-volume interface values $\eta_{2.5}$, $\eta_{3.5}$ are evaluated using cells 1 to 5). However, if the first cell's control-volume interface values $\eta_{0.5}$, $\eta_{1.5}$ are evaluated using cells 1 to 5, unpredictable behavior might appear. On the other hand, although a lower order of accuracy of the interpolation scheme at the boundaries does not affect the overall accuracy in the full domain, if cell 1 uses a uniform distribution of η, such that $\eta_{0.5} = \eta_1 = \eta_{1.5}$, the numerical diffusion might be too large and it might mask or filter out the waves at the boundaries. Conservatively, we use a central three-point interpolation for cell 2:

$$\eta_{1.5} = \frac{1}{3}\eta_1 + \frac{5}{6}\eta_2 - \frac{1}{6}\eta_3, \quad (21)$$

$$\eta_{2.5} = -\frac{1}{6}\eta_1 + \frac{5}{6}\eta_2 + \frac{1}{3}\eta_3, \quad (22)$$

and a one-side interpolation for cell 1:

$$\eta_{0.5} = \frac{3}{2}\eta_1 - \frac{1}{2}\eta_2, \quad (23)$$

$$\eta_{1.5} = \frac{1}{2}\eta_1 + \frac{1}{2}\eta_2. \quad (24)$$

This treatment is not the least diffusive, but it is more generally applicable.

We derive the LMARS for the vertical Lagrangian coordinate in sections 3a and 3b, provide the calculation

using LMARS for the Eulerian coordinate and some tuning techniques in sections 3c and 3d. The boundary conditions, time integration, and the remapping scheme are discussed in sections 3e, 3f, and 3g.

a. Horizontal Riemann solver

To derive the interface flux vector **F** in the first step, we need $u_{1/2}$ and $\Psi_{1/2}$.

When evaluating the fluxes at each cell face, only one flux vector at each cell face, namely, the vector of normal fluxes is needed. The vector of normal fluxes can be obtained by evaluating the normal speed $u_{1/2}$ and the pseudopressure $\Psi_{1/2}$ using a local one-dimensional form of the Euler equation. Equation (3) can be written in the form of an advection-type equation, assuming a zero right-hand side (rhs) for purely horizontal flow:

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = -\frac{1}{\pi}\frac{\partial \Psi}{\partial x}. \tag{25}$$

Assuming a local isothermal condition, we used the gas pressure equation in the form generally defined in compressible flow (Laprise 1992):

$$\frac{dp}{dt} + a^2\rho\frac{\partial u}{\partial x} = 0, \tag{26}$$

where $a = \sqrt{\gamma p/\rho} = \sqrt{\gamma\Psi/\pi}$ is the Eulerian speed of sound. Taking the total derivative of Eq. (4) and using Eqs. (26) and (2) we have

$$\frac{\partial \Psi}{\partial t} + u\frac{\partial \Psi}{\partial x} = -a^2\pi\frac{\partial u}{\partial x}. \tag{27}$$

Note that Eqs. (25) and (27) do not need to be solved using the conservation form of the equations, since they are only used for deriving an expression to calculate the values of $u_{1/2}$ and $\Psi_{1/2}$ on the interfaces of the FV cells in the x direction. In general, these two equations can be represented as

$$\mathbf{U}_t + \mathbf{A}\mathbf{U}_x = 0, \tag{28}$$

$$\text{where} \quad \mathbf{U} = \begin{pmatrix} u_{1/2} \\ \Psi_{1/2} \end{pmatrix}; \quad \text{and} \quad \mathbf{A} = \begin{pmatrix} u_{1/2} & 1/\pi \\ a^2\pi & u_{1/2} \end{pmatrix}. \tag{29}$$

Solving this system, $\mathbf{AU} = \lambda\mathbf{U}$, for its eigenvalues, we find that $\lambda_{1,2} = u_{1/2} \mp a$.

We assume that we have a discontinuity at the interface of two FV cells, which comes from the interpolation of u and Ψ in different FV cells. Using the interpolation scheme given in Eqs. (19) and (20), we define the pair of the variables u and Ψ to the left of the discontinuity as $\mathbf{U}_l = [u_l, \Psi_l]$, to the right of it as $\mathbf{U}_r = [u_r, \Psi_r]$, and at the interface as $\mathbf{U} = [u_{1/2}, \Psi_{1/2}]$. Then we use the Rankine–Hugoniot "jump" conditions (Hirsh 2007) to write the following:

$$\lambda_1(\mathbf{U} - \mathbf{U}_l) = \mathbf{A}(\mathbf{U} - \mathbf{U}_l) \quad \text{and}$$
$$\lambda_2(\mathbf{U} - \mathbf{U}_r) = \mathbf{A}(\mathbf{U} - \mathbf{U}_r). \tag{30}$$

For $\lambda_1 = u_{1/2} - a$ we have

$$\begin{pmatrix} u_{1/2} & 1/\pi \\ a^2\pi & u_{1/2} \end{pmatrix}\begin{pmatrix} u_{1/2} - u_l \\ \Psi_{1/2} - \Psi_l \end{pmatrix} = (u_{1/2} - a)\begin{pmatrix} u_{1/2} - u_l \\ \Psi_{1/2} - \Psi_l \end{pmatrix}.$$
$$\text{or} \quad \Psi_{1/2} + \pi a u_{1/2} = \Psi_l + \pi a u_l \tag{31}$$

Similarly, for $\lambda_2 = u_{1/2} + a$:

$$\begin{pmatrix} u_{1/2} & 1/\pi \\ a^2\pi & u_{1/2} \end{pmatrix}\begin{pmatrix} u_{1/2} - u_r \\ \Psi_{1/2} - \Psi_r \end{pmatrix} = (u_{1/2} + a)\begin{pmatrix} u_{1/2} - u_r \\ \Psi_{1/2} - \Psi_r \end{pmatrix}.$$
$$\text{or} \quad \Psi_{1/2} - \pi a u_{1/2} = \Psi_r - \pi a u_r \tag{32}$$

Rearranging Eqs. (31) and (32), we obtain the values of velocity u and Ψ at interfaces of the cells:

$$\Psi_{1/2} = \frac{1}{2}(\Psi_r + \Psi_l) - \frac{\pi a}{2}(u_r - u_l), \tag{33}$$

$$u_{1/2} = \frac{1}{2}(u_r + u_l) - \frac{1}{2\pi a}(\Psi_r - \Psi_l). \tag{34}$$

The flux **F** is updated in the second step according to Eq. (18).

b. Vertical Riemann solver

To fully solve the system of equations represented by Eq. (17), we need to calculate the source vector $\mathbf{S} = [0, -(p\partial\Phi/\partial x)_{1/2}, gp'_{1/2}, 0]$. The source terms are discretized in the vertical direction, and the discretization terms $-p\partial\Phi/\partial x$ and gp' are defined at the layer interfaces. Here, the subscript ½ denotes to the value at the vertical layer interfaces. Thus, we need to find the values of $p'_{1/2}$, $w_{1/2}$, and $\Phi_{1/2}$ at the vertical interfaces of the Lagrangian layers. Although there is no vertical flux across the Lagrangian coordinate, we can still create a Riemann problem to derive an expression to calculate the values of $w_{1/2}$ and $p'_{1/2}$ at the vertical interfaces by performing the first step of LMARS, so that $\Phi_{1/2}$ can be updated using Eq. (7).

Similar to the treatment for the horizontal flux evaluation, the local normal velocity and pressure at the control-volume interface can be evaluated using the one-dimensional Euler equations. We start by differentiating the state equation in Eq. (8) along the time axis. Then,

taking into account Eqs. (1) and (6), where, for pure vertical flow, π and Θ are both constant, we derive

$$\frac{\partial p}{\partial t} = \gamma\rho p \frac{\partial w/\partial s}{\partial m/\partial s}, \tag{35}$$

where $\partial m/\partial s = -\rho\partial z/\partial s$. Using the definition $p' = p - p^*$, we assume $\partial p^*/\partial t = 0$. This is valid because mass is conserved in each control volume, and we can write the following:

$$\frac{\partial p'}{\partial t} - C^2\frac{\partial w/\partial s}{\partial m/\partial s} = 0; \quad \text{or} \quad \frac{\partial p'}{\partial t} - C^2\frac{\partial w}{\partial m} = 0, \tag{36}$$

where $C = \sqrt{\gamma\rho p}$ is the sound speed along the m axis in the Lagrangian vertical coordinate. From Eqs. (1) and (5) and considering only vertical movements, $(\partial/\partial x = 0)$, we obtain

$$\frac{\partial w}{\partial t} - \frac{\partial p'/\partial s}{\partial m/\partial s} = 0; \quad \text{or} \quad \frac{\partial w}{\partial t} - \frac{\partial p'}{\partial m} = 0. \tag{37}$$

Equations (36) and (37) will be used to solve the Riemann problem at the interface of the Lagrangian layers. Analogous to the horizontal case, we have

$$\mathbf{U}_t + \mathbf{A}\mathbf{U}_m = 0, \tag{38}$$

$$\text{where} \quad \mathbf{U} = \begin{pmatrix} w \\ p' \end{pmatrix}; \quad \text{and} \quad \mathbf{A} = \begin{pmatrix} 0 & -1 \\ -C^2 & 0 \end{pmatrix}. \tag{39}$$

As before, we find the eigenvalues: $\lambda_{1,2} = \mp C$. Also, we assume that we have a discontinuity of w and p' at the interface of two Lagrangian layers, and we define the pair of the variables w and p' above the discontinuity as $\mathbf{U}_u = [w_u, p'_u]$, below it as $\mathbf{U}_d = [w_d, p'_d]$, (where u, d denote up and down), and at the interface as $\mathbf{U}_d = [w_{1/2}, p'_{1/2}]$. Then, the Rankine–Hugoniot conditions are

$$\begin{aligned} &\lambda_1(\mathbf{U} - \mathbf{U}_u) = \mathbf{A}(\mathbf{U} - \mathbf{U}_u) \quad \text{and} \\ &\lambda_2(\mathbf{U} - \mathbf{U}_d) = \mathbf{A}(\mathbf{U} - \mathbf{U}_d). \end{aligned} \tag{40}$$

Equation (40) can be expanded for both eigenvalues as

$$\begin{pmatrix} 0 & -1 \\ -C^2 & 0 \end{pmatrix}\begin{pmatrix} w - w_u \\ p'_{1/2} - p'_u \end{pmatrix} = -C\begin{pmatrix} w_{1/2} - w_u \\ p'_{1/2} - p'_u \end{pmatrix}$$
$$\text{or} \quad p'_{1/2} - Cw_{1/2} = p'_u - Cw_u \tag{41}$$

$$\begin{pmatrix} 0 & -1 \\ -C^2 & 0 \end{pmatrix}\begin{pmatrix} w_{1/2} - w_d \\ p'_{1/2} - p'_d \end{pmatrix} = C\begin{pmatrix} w_{1/2} - w_d \\ p'_{1/2} - p'_d \end{pmatrix}$$
$$\text{or} \quad p'_{1/2} + Cw_{1/2} = p'_d + Cw_d. \tag{42}$$

Rearranging Eqs. (41) and (42), we obtain

$$p'_{1/2} = \frac{1}{2}(p'_d + p'_u) + \frac{C}{2}(w_d - w_u), \tag{43}$$

$$w_{1/2} = \frac{1}{2}(w_d + w_u) + \frac{1}{2C}(p'_d - p'_u). \tag{44}$$

The overall stability is not sensitive to the representation of the horizontal velocity at the interface. The simplest representation $u_{1/2} = (u_d + u_u)/2$, is sufficient. With $u_{1/2}$ and $w_{1/2}$, the geopotential $\Phi_{1/2}$ can be updated using Eq. (7). These values can then be used to calculate the source vector $\mathbf{S}$.

c. *The LMARS in Eulerian coordinate*

The system of equations in the Eulerian coordinate is

$$\frac{\partial \mathbf{R}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{H}}{\partial z} = 0, \tag{45}$$

where $\mathbf{R} = (\rho, \rho u, \rho w, \tilde{\Theta})$, $\mathbf{F} = (\rho u, \rho uu + p, \rho wu, \tilde{\Theta}u)$, and $\mathbf{H} = (\rho w, \rho wu, \rho ww + p', \tilde{\Theta}w)$, with $\tilde{\Theta} = \rho\theta/p_0^{\kappa}$. Similar to the representation in the Lagrangian system, we derive the left and right flux vectors as

$$\mathbf{F} = \mathbf{R}_b u_{1/2} + p_{1/2}(0,1,0,0), \tag{46}$$

$$\mathbf{H} = \mathbf{R}_c w_{1/2} + p'_{1/2}(0,0,1,0), \tag{47}$$

where

$$b = \begin{cases} l, & \text{if} \quad u_{1/2} > 0 \\ r, & \text{if} \quad u_{1/2} \le 0 \end{cases} \quad \text{and} \quad c = \begin{cases} d, & \text{if} \quad w_{1/2} > 0 \\ u, & \text{if} \quad w_{1/2} \le 0 \end{cases}. \tag{48}$$

The derivation of the LMARS is very similar to that given in section 3a. Here we provide the result:

$$p_{1/2} = \frac{1}{2}(p_r + p_l) - \frac{\rho a}{2}(u_r - u_l), \tag{49}$$

$$u_{1/2} = \frac{1}{2}(u_r + u_l) - \frac{1}{2\rho a}(p_r - p_l), \tag{50}$$

$$p'_{1/2} = \frac{1}{2}(p'_d + p'_u) + \frac{\rho a}{2}(w_d - w_u), \tag{51}$$

$$w_{1/2} = \frac{1}{2}(w_d + w_u) + \frac{1}{2\rho a}(p'_d - p'_u). \tag{52}$$

Although the system represented by Eqs. (17) and (45) is only described in 2D, the extension of the solution to multidimensions follows the standard procedure using directional splitting.

d. The tuning of LMARS

Equations (33), (34), (43), (44), and (49)–(52) are in similar form, if we denote $\tilde{p}$, $\tilde{\rho}$, $\tilde{u}$, and $\tilde{a}$ as the pressure/pseudopressure, density/pseudodensity, velocity perpendicular to the flux interface, and speed of sound, respectively. The equations for the velocity and pressure at the flux interface can be expressed as

$$\tilde{p}_{1/2} = \frac{1}{2}(\tilde{p}_{\text{upwind}} + \tilde{p}_{\text{downwind}}) + \frac{\tilde{\rho}\tilde{a}}{2}(\tilde{u}_{\text{upwind}} - \tilde{u}_{\text{downwind}}), \tag{53}$$

$$\tilde{u}_{1/2} = \frac{1}{2}(\tilde{u}_{\text{upwind}} + \tilde{u}_{\text{downwind}}) + \frac{1}{2\tilde{\rho}\tilde{a}}(\tilde{p}_{\text{upwind}} - \tilde{p}_{\text{downwind}}). \tag{54}$$

The second terms on the right-hand side are associated with implicit diffusion effects. This diffusion can be minimized under conditions, which do not have strong vertical convection and have continuous physical variables with small perturbations, by introducing the following form:

$$\tilde{p}_{1/2} = \frac{1}{2}(\tilde{p}_{\text{upwind}} + \tilde{p}_{\text{downwind}}) + \beta\frac{\tilde{\rho}\tilde{a}}{2}(\tilde{u}_{\text{upwind}} - \tilde{u}_{\text{downwind}}), \tag{55}$$

$$\tilde{u}_{1/2} = \frac{1}{2}(\tilde{u}_{\text{upwind}} + \tilde{u}_{\text{downwind}}) + \beta\frac{1}{2\tilde{\rho}\tilde{a}}(\tilde{p}_{\text{upwind}} - \tilde{p}_{\text{downwind}}), \tag{56}$$

where β is a variable diffusion parameter, $\beta \leq 1$. Using $\beta = 1$ provides stability for most situations, but introduces more diffusion. Using a smaller value of β, we are able to achieve smaller diffusion in the LMARS scheme, but this should be tested on a case-by-case basis.

e. Boundary conditions

1) HORIZONTAL BOUNDARY CONDITIONS

Since all the prognostic variables are defined on an unstaggered grid, we can apply standard boundary conditions in the horizontal direction (the position of Φ is vertically staggered, however, it is unstaggered in the horizontal direction). In the bubble tests shown in section 4, a reflective boundary condition is applied by mirroring the ghost cells. The ghost cells are extra grid cells on the boundaries used for the interpolation of the variables.

In the gravity wave test shown in section 4, a periodic boundary condition is applied.

2) VERTICAL BOUNDARY CONDITIONS

In the vertical Lagrangian coordinate, the top/bottom boundary conditions are derived from the Eqs. (41) and (42). At the bottom of the model we adopt a reflective surface by setting

$$w = 0, \tag{57}$$

$$p' = p'_u - Cw_u. \tag{58}$$

At the top of the model, we have two options: either a ''rigid lid'' condition or an ''open boundary'' condition to allow waves and disturbances originating within the model domain to leave the domain without affecting the interior solution.

For the rigid lid condition, we set reflective conditions similar to the bottom boundary condition:

$$w = 0. \tag{59}$$

$$p' = p'_d + Cw_d, \tag{60}$$

For the open boundary condition we set these as

$$w = w_d. \tag{61}$$

$$p' = p'_d, \tag{62}$$

Although we do not have any vertically oriented ghost cells, however, using Eqs. (43)–(44) and (61)–(62) we can derive that the upper and lower side values at the boundary interface are identical, which has an effect similar to that of building a ghost cell using the nonreflective boundary condition.

The vertical boundary conditions in the Eulerian coordinate are all set as reflective boundary conditions.

f. Time integration

The application of a two-step prediction-correction time marching scheme with the conservative five-point polynomial interpolation scheme, will lead to different results if different time steps are used, because the two-step prediction-correction time marching scheme is only second-order accurate in time (unless very small time steps are used). However, when paring the conservative five-point polynomial interpolation scheme with a four-step Runge–Kutta method, the choice of time step would not affect the result as long as the CFL < 1 condition is met. In two dimensions, the interface values may be regarded as having been averaged along the interface. The interface flux based on such an average, though, is not the proper flux average along the interface, because the flux is not a linear function of the state quantities. So fourth-order accuracy is downgraded to second-order accuracy. However, the higher-order interpolation scheme does achieve low diffusivity of the overall scheme so that we are able to observe small-scale structures if we use a small grid size configuration.

We use a four-step Runge–Kutta method to integrate the equation for the prognostic variables in time. In the

nonhydrostatic Lagrangian version, the prognostic variables are $\mathbf{U} = (\pi, u, w, \Theta, \Phi)$ [Eq. (7) is included, which is not in flux form]. The prognostic variable vectors in the Eulerian version and the hydrostatic version are similar. The differential form of $\mathbf{U}$ is

$$\frac{d\mathbf{U}}{dt} = \mathrm{RHS}(\mathbf{U}), \tag{63}$$

where RHS stands for ''right hand side'' of the system of differential Eqs. (1), (3), (5), (6), and (7), and these functions are independent with variable of time . The fourth-order in time four-step Runge–Kutta method is

$$\mathbf{U}^1 = \mathbf{U}(t) + \frac{\Delta t}{2}\mathrm{RHS}[\mathbf{U}(t)], \tag{64}$$

$$\mathbf{U}^2 = \mathbf{U}(t) + \frac{\Delta t}{2} \times \mathrm{RHS}(\mathbf{U}^1), \tag{65}$$

$$\mathbf{U}^3 = \mathbf{U}(t) + \Delta t \times \mathrm{RHS}(\mathbf{U}^2), \tag{66}$$

$$\mathbf{U}(t + \Delta t) = \mathbf{U}(t) + \frac{\Delta t}{6}\{\mathrm{RHS}[\mathbf{U}(t)] + 2\mathrm{RHS}(\mathbf{U}^1) + 2\mathrm{RHS}(\mathbf{U}^2) + \mathrm{RHS}(\mathbf{U}^3)\}, \tag{67}$$

where Δt is the time step. More standard four-step Runge–Kutta methods are described in Durran (2010).

g. Vertical remap

When using the vertical Lagrangian coordinate, the Lagrangian surfaces that bound an atmospheric layer deform and need to be remapped onto the original coordinates. The volume mean prognostic variables π, πu, πw, and Θ are remapped. Taking π as an example, our procedure is as follows: 1) Integrate π in the top-down direction to build a continuous profile. If we denote the vertical integral to grid level $k + 0.5$ as $m_{k+0.5} = \sum_{i=1}^{k} \pi_i$, with $m_{0.5} = 0$, then $m_{k+0.5}$ is defined at the layer interface with vertical location represented by the geopotential $\Phi = gz$, where z is the height. 2) The profile of $m_{k+0.5}$ is acquired using a seven-point polynomial interpolation scheme, with a Newton form polynomial interpolation (Yang 2001), that is, to get m at location $\Phi_{\mathrm{ref}_{k+0.5}}$ [i.e., $(m_{\mathrm{new}_{k+0.5}}, \Phi_{\mathrm{ref}_{k+0.5}})$], we use the following inputs to the polynomial interpolation $(m_{k-2.5}, \Phi_{k-2.5})$, $(m_{k-1.5}, \Phi_{k-1.5})$, $(m_{k-0.5}, \Phi_{k-0.5})$, $(m_{k+0.5}, \Phi_{k+0.5})$, $(m_{k+1.5}, \Phi_{k+1.5})$, $(m_{k+2.5}, \Phi_{k+2.5})$, and $(m_{k+3.5}, \Phi_{k+3.5})$. Near the top and bottom of the domain, the topmost and bottommost seven points are used for the input to the polynomial interpolation [i.e., $(m_{\mathrm{new}_{1.5}}, \Phi_{\mathrm{ref}_{1.5}})$, $(m_{\mathrm{new}_{2.5}}, \Phi_{\mathrm{ref}_{2.5}})$, $(m_{\mathrm{new}_{3.5}}, \Phi_{\mathrm{ref}_{3.5}})$ are all calculated using the profile built by the points $(m_{0.5}, \Phi_{0.5})$, $(m_{1.5}, \Phi_{1.5})$, $(m_{2.5}, \Phi_{2.5})$, $(m_{3.5}, \Phi_{3.5})$, $(m_{4.5}, \Phi_{4.5})$, $(m_{5.5}, \Phi_{5.5})$, $(m_{6.5}, \Phi_{6.5})$ using a polynomial interpolation]. 3) Using the profile $(m_{\mathrm{new}_{k+0.5}}, \Phi_{\mathrm{ref}_{k+0.5}})$, the remapped π is calculated from $\pi_{\mathrm{new}_k} = m_{\mathrm{new}_{k+0.5}} - m_{\mathrm{new}_{k-0.5}}$. Steps 1 to 3 $(m_{\mathrm{new}_{k+0.5}}, \Phi_{\mathrm{ref}_{k+0.5}})$ complete the remap for π. Since the values of $m_{k+0.5}$ at the top and bottom remain unchanged during the remapping process, the total mass $\Sigma\pi/g$ is automatically conserved. Similarly, the total momentum is also conserved (conservation of $\Sigma\pi u$ and $\Sigma\pi w$), and no extra heat is introduced into the system (conservation of $\Sigma\Theta$). For simplicity, there is no limiting mechanism used in step 2.

4. Tests and results

a. Robert's warm bubble tests

We test our nonhydrostatic approach for solving the equations for the 2D (x, z) nonhydrostatic atmosphere using two standard tests from Robert (1993). These tests are for two different types of warm bubbles: the ''Gaussian'' and ''uniform'' bubbles, which rise in an isentropic atmosphere (303.15 K) within a closed box.

The Gaussian bubble is placed in a 1 km wide by 1.5-km-high box and is represented by a perturbation of the potential temperature of the following form:

$$\theta' = \begin{cases} A, & \text{if} \quad r \le a \\ Ae^{-(r-a)^2/s^2}, & \text{if} \quad r > a \end{cases}, \tag{68}$$

where $r^2 = (x - x_0)^2 + (z - z_0)^2$ is the distance from the bubble center, $x_0 = 500$ m, $z_0 = 260$ m, $A = 0.5$ K, $a = 50$ m, and $s = 100$ m. The uniform bubble with a radius of 250 m and with an initial 303.65 K ''flat'' potential temperature was positioned at $(x_0, z_0) = (500, 260)$ m in a 1 km by 1 km box.

1) Eulerian framework versus Lagrangian framework

Although the Lagrangian framework is useful in reducing the 2D flow to 1D flow, and hence enhancing the computational efficiency of the solution, it allows the finite-volume cells to deform from their rectangular shape, which will introduce some geometric errors when the Lagrangian interfaces of the layers are significantly distorted. Using the Eulerian framework avoids this problem because the rectangular shape of the grid cells is fixed. The LMARS solver can calculate fluxes in both Lagrangian coordinates and Eulerian coordinates using a similar framework, so the results in the Eulerian framework can serve as the reference solution.

Figure 1 shows the potential temperature perturbation in the experiments. The output times selected are the same as those shown in Robert (1993). The first row

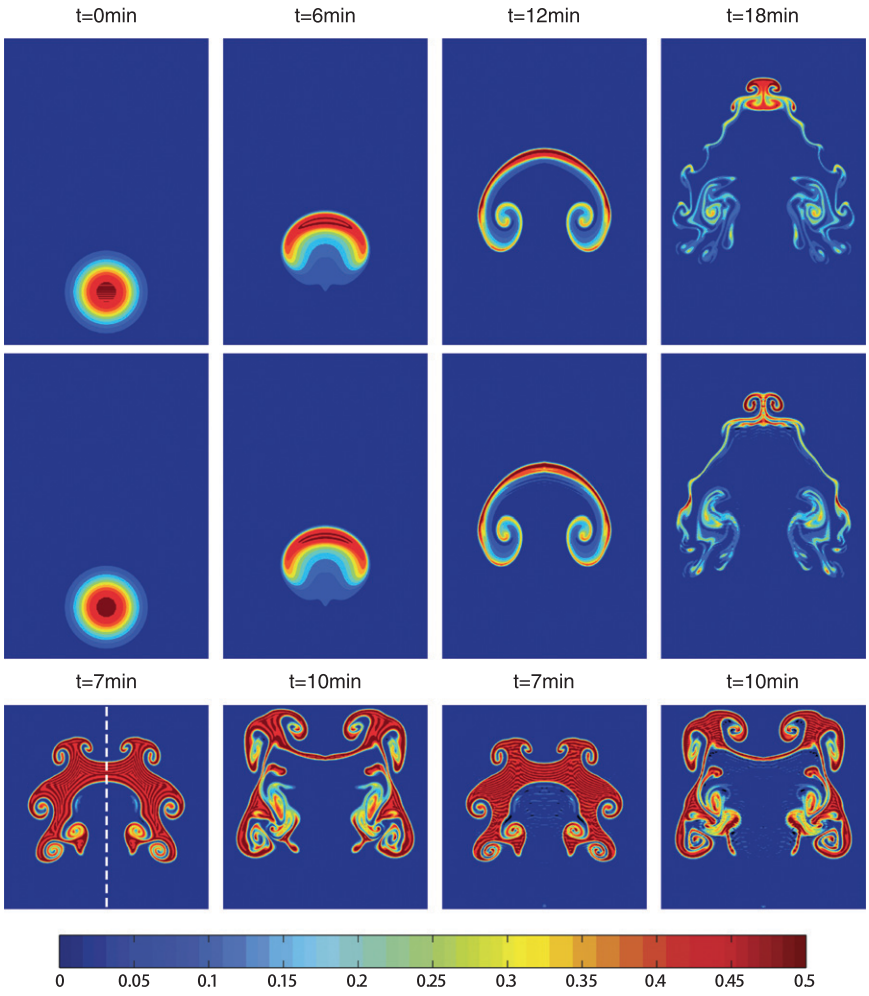

FIG. 1. (top two rows) The potential temperature (PT) (K) for a Gaussian bubble perturbation in a 1 km by 1.5 km domain using (top) the Eulerian coordinate and (middle)) the Lagrangian coordinate: (left to right) t = 0–18 min. (bottom) An initial uniform bubble perturbation in a 1 km by 1 km domain using (left two panels) the Eulerian coordinate and (right two panels) the Lagrangian coordinate: (from left to right) t = 7–10 min. The grid spacing of all results is 5 m. The cross section of the PT perturbation along the dashed line in the bottom-left panel is presented in Fig. 2.

shows the Gaussian bubble results from the Eulerian configuration, while the second row shows the same results from the Lagrangian configuration. The output times for the Gaussian bubble results are 0, 6, 12, and 18 min. The last row shows the uniform bubble test, with the left two subplots from the Eulerian configuration while the right two are from the Lagrangian configuration. The grid size for all the results was $\Delta x = \Delta z = 5$ m, and the time step $\Delta t = 0.007$ s scales with the grid spacing. The color range is presented from 0 to 0.5 K for comparison purposes with the results shown by Robert (1993). The output times for the uniform bubble results are at 7 and 10 min. The AUSM$^+$-up method was also tested with the Eulerian configuration and shows almost identical results (figures omitted), but requires about 50% more computer time compared to the LMARS method.

Since no limiter is applied in either interpolation scheme for the variables or remapping, two grid-size waves can be observed. These waves do not grow or cause instability. Figure 2 provides a clearer picture of theses small-scale oscillations. It shows the cross section at the center of the uniform bubble test for the 7-min plot (the dashed line in the bottom-left plot in Fig. 1). The oscillations are especially present near the sharp edges of the rising uniform bubble.

In Fig. 1, the 6- and 12-min results for the two different coordinate configurations agree well with each other

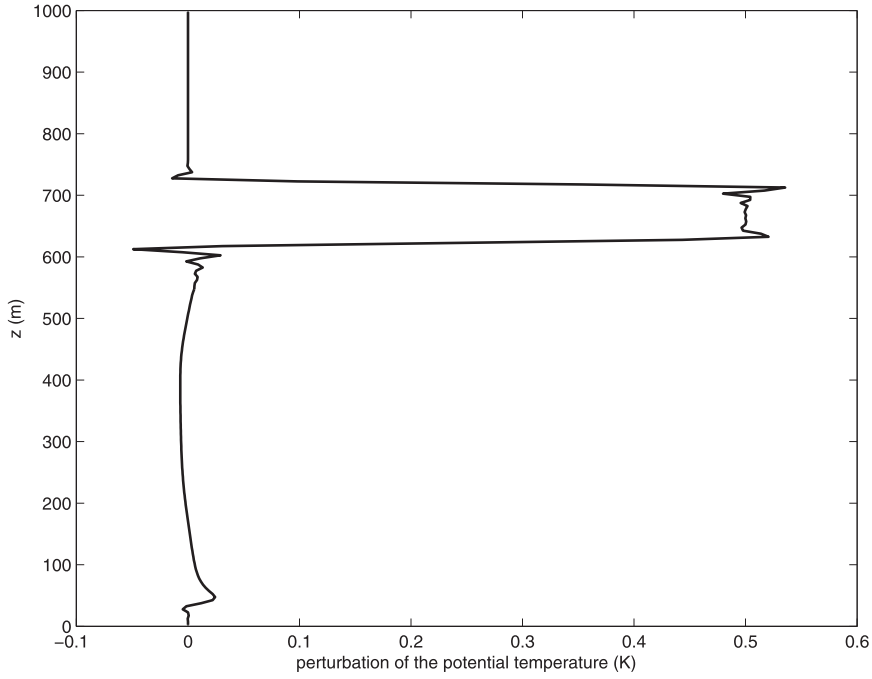

FIG. 2. Cross section of the potential temperature perturbation (K) at $x = 500$ m of the uniform bubble test after 7 min using the Eulerian coordinate.

and with the corresponding solutions by Robert (1993). However, at 18 min, the Kelvin–Helmholtz instability starts to appear, and the results are different between the two different schemes, especially for the shape of the bubble head. Since we have not applied any diffusion or limiters in our experiments, the numerical solutions are only expected to agree at finite times and may diverge depending on the implicit damping and dispersion characteristics of the chosen algorithms. Similar behavior is also found when comparing the results of the uniform bubble: the general shapes of the uniform bubbles are similar; however, the Kelvin–Helmholtz instability is resolved differently between different schemes at longer times. The bubble head and bubble's width show a slight difference between our results and those of Robert (1993), which could be due to different applications of the reflective boundary conditions.

In this particular case, because the Eulerian framework is able to maintain a rectangular shape for the control volume at each time step, the fluxes are always perpendicular to the control-volume interfaces. Moreover, we note that the error induced by the remapping scheme is not included in the Eulerian framework. Therefore, we consider the Eulerian version to be the more accurate solution and use it as the reference below. In section 5a, we also conducted a sensitivity test in which we added strong viscosity to both the Lagrangian and Eulerian formulations. With the addition of viscosity, the results for both formulations converge at high resolution. But at coarse resolution, the plots using the Eulerian formulation are slightly closer to the high-resolution solution. In general, if subgrid turbulence is added to a model using a parameterization, both Eulerian and Lagrangian frameworks should provide results that converge. At shorter time scales, before any turbulence develops in the Gaussian bubble tests (i.e., at 6 and 12 min) or for the results of the gravity wave tests in the next section, the Eulerian and Lagrangian results are similar. The difference between the Eulerian and Lagrangian results without explicit diffusion, which related to the plots mentioned above are compared in section 5c.

2) GRID SPACING VERSUS NUMERICAL DIFFUSION

We do not need to apply any diffusion such as divergence damping to stabilize our numerical scheme because our LMARS solver provides the necessary stability. Since LMARS is applied in both the horizontal and vertical directions, any implicit numerical diffusion is consistent in all directions. As the diffusion is determined by the difference of the pressure and velocities at the interface of the FV cells, which are acquired by interpolation, a high-order interpolation scheme will lead to less diffusion.

The numerical diffusion is not a linear function of the grid size. As a result, when the grid spacing is decreased,

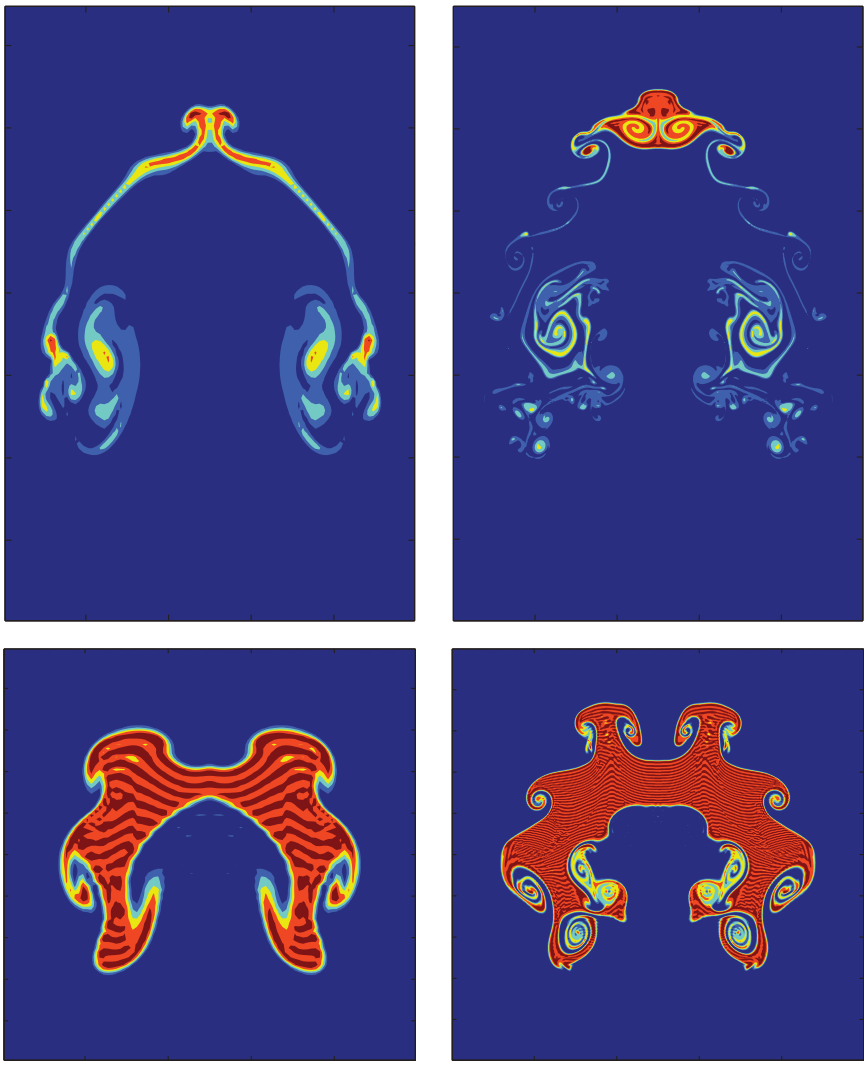

FIG. 3. Plot of the Gaussian bubble at 18 min using (top) the Eulerian configuration and (bottom) the result of the uniform bubble at 7 min in the Lagrangian configuration with a grid size of (left) 10 and (right) 2.5 m.

the result may not converge. Instead, numerical viscosity will be decreased strongly and an ultra-low diffusive result will be found. Figure 3 shows a plot at 18 min of the Gaussian bubble using the Eulerian configuration and the result of the uniform bubble at 7 min in the Lagrangian configuration with a grid size of 10 and 2.5 m, respectively. The 10-m run is actually more diffusive than the results of Robert (1993), however, the 2.5-m results are of high quality. With smaller grid size, the smaller scale Kelvin–Helmholtz waves can be resolved in the Gaussian bubble. In addition, in the uniform bubble test, the discontinuity at the bubble edge is very sharp.

These tests show that we are able to resolve the warm bubble tests with small numerical diffusion using a high-order interpolation scheme for the prognostic variables and a small grid size. Although the amount of numerical diffusion that can be tolerated in the solution depends on the specific application, and for some applications, an economic computational performance is preferred, we have shown that LMARS provides stable solutions with a small amount of implicit numerical diffusion. Our algorithm can be ''downgraded'' by using a lower-order interpolation scheme for the prognostic variables without any special treatment such as introducing a divergence damping term for stability. An implicit time marching scheme might also be used and equipped with the LMARS numeric solver to filter out acoustic waves and achieve a Courant–Friedrichs–Lewy (CFL) number larger than one, but this technique will

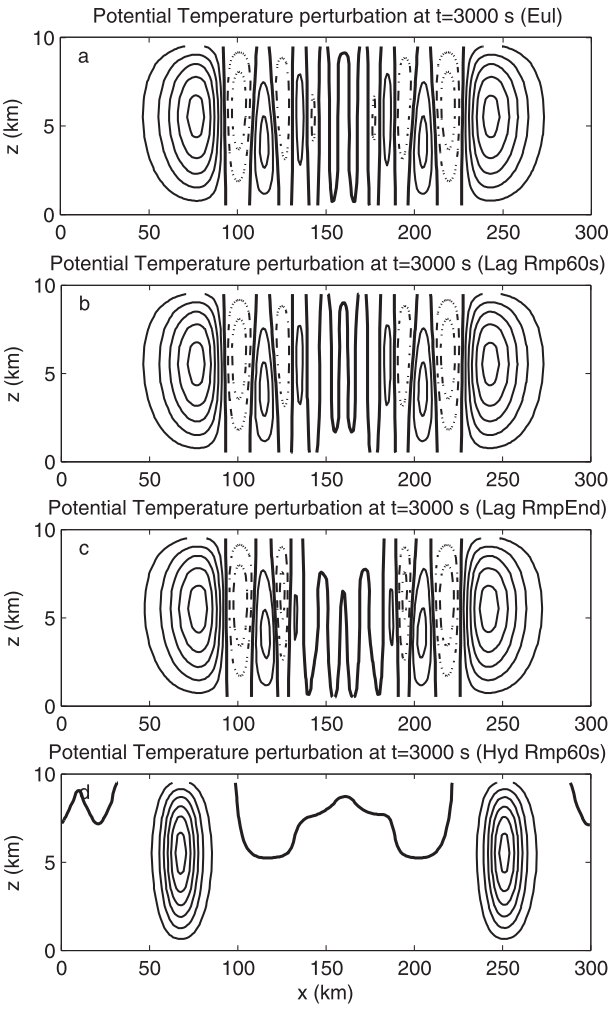

FIG. 4. The potential temperature perturbation (K) at $t = 3000$ s in the gravity wave test. The configurations are (a) nonhydrostatic with Eulerian coordinates; (b) nonhydrostatic with vertical Lagrangian coordinate and a remap frequency of 60 s; (c) as in (b), but with only a single remap at the end of the simulation; and (d) hydrostatic with vertical Lagrangian coordinates with 60-s remap frequency. The contour interval is 0.0005 K, the bold line is the 0 contour, the solid lines are positive, and dash–dotted lines are negative. The grid spacing is $dx = dz = 1$ km.

introduce additional diffusion into the solution. The discussion of time-marching schemes is beyond the scope of this work.

b. Gravity wave test

Figure 4 presents the results from a gravity wave test using the vertical Lagrangian coordinate, Eulerian coordinates, with the nonhydrostatic and hydrostatic setups, which is very similar to the test case developed by Skamarock and Klemp (1994). The vertical domain size is 10 km and the horizontal domain size is 300 km. The background atmosphere has a constant Brunt–Väisälä frequency of 10^{-2} s^{-1} with a surface temperature of 300 K, and the surface pressure is 10^5 Pa. The grid spacings are $\Delta x = \Delta z = 1$ km and the time step is 2 s. The initial horizontal wind is 20 m s^{-1} and a periodic horizontal boundary condition is used. The waves are excited by an initial θ perturbation of the following form:

$$\theta' = \Delta\theta_0 \frac{\sin\dfrac{\pi z}{H}}{1 + (x - x_c)^2/a^2}, \tag{69}$$

where $\Delta\theta_0 = 10^{-2}$ K, $H = 10$ km, $a = 5$ km, and $x_c = 100$ km.

The original test published in Skamarock and Klemp (1994) used a Boussinesq model and a rigid lid boundary condition. A rigid lid boundary condition is used in the test using the Eulerian coordinates. However, the way we apply the hydrostatic approximation requires that we use a free surface at the top of the model, so an open boundary condition is applied in all Lagrangian coordinate tests. Additionally, since the Coriolis force would bring in one more equation for v, and this would break the momentum conservation, we did not include this in our test, and set the Coriolis parameter to zero. This is different from the approach in Skamarock and Klemp (1994). However, Giraldo and Restelli (2008) conducted a similar test, without the Coriolis force, with the full Euler equations. The perturbation potential temperature at $t = 3000$ s is shown in Fig. 4 for comparison with the nonhydrostatic results of Giraldo and Restelli (2008). Both the hydrostatic and the nonhydrostatic simulations start from identical initial conditions.

Figure 4a is the potential temperature perturbation using the nonhydrostatic configuration in the Eulerian coordinates. This result is in good agreement with the results of Giraldo and Restelli (2008) who used a local spectral method. Our result is slightly more damping, which may due to the fact that Giraldo and Restelli (2008) used a 250-m resolution and 10th-order polynomials. Figures 4b,c present the results of the nonhydrostatic configuration in the vertical Lagrangian coordinate. The remap frequency of the result in Fig. 4b is every 60 s, while that in Fig. 4c is only remapped at the end of the simulation. Figure 4d uses a hydrostatic configuration with remapping every 60 s.

This test is dominated by the evolution of the gravity wave. The horizontal background velocity of the fluid, which is close to the gravity wave speed, is much greater than the vertical velocity. So even with different boundary conditions at the top of the model, there is no visible deformation at the model top, and the reflection effect is small. For the same reason, the FV vertical deformation is very small and the difference due to different

remapping frequencies in this test is negligible. The only differences are the zero-contour differences in the middle of the domain. The model's Eulerian version does not involve remapping, so it is least diffusive. However, since the model's Lagrangian version translates the vertical motion into the finite-volume deformation, and it has no vertical flux terms to calculate, it is more computationally efficient than the Eulerian version.

Figure 4d is the gravity wave simulation using the hydrostatic formulations in the vertical Lagrangian coordinate. The same grid spacing and time step is used. Since the hydrostatic formulations does not permit vertical acoustic wave, the dispersive wave trains observed in the nonhydrostatic model are not generated in the hydrostatic mode.

5. Quantitative validation

a. Comparison of the Eulerian and Lagrangian formulations with large viscosity for the Gaussian warm bubble test

LMARS can be used in both the Eulerian and Lagrangian frameworks. To demonstrate that both methods converge to a similar result, we added a second-order explicit damping term to the momentum equation.

We repeated the rising Gaussian warm bubble test from Robert (1993), with a strong diffusion term of the form $\pi K(u_{xx} + u_{zz})$ and $\pi K(w_{xx} + w_{zz})$ added on the right-hand side of the horizontal and vertical momentum equations to represent the addition of a term with large viscosity. The diffusion coefficient K was set to a value of 75 $m^2\ s^{-1}$. We compared the results at different resolutions: $dx = dz = 5$, 10, 20, and 40 m and at the time $t = 18$ min. The vertical domain size is extended to 1520 m when the grid spacing of 40 m is used. The results are illustrated in Fig. 5.

Because the diffusion term changes the viscosity of the air, the Kelvin–Helmholtz instability is not observed.

The Eulerian and Lagrangian formulations started to have converged results at a resolution of $dx = dz =$ 10 m. When the grid spacing increases, the quality of the results is degraded in both the Eulerian and Lagrangian versions. In particular, the Eulerian version maintains ''smoother'' results than does the Lagrangian version. Thus, we judge that the Eulerian formulation produces results that are in better agreement with the converged solution at high resolution than is the Lagrangian version. In our simulations, the results calculated using the Eulerian formulation are used as the reference solution.

b. Order of accuracy analysis

The advection terms in our numerical scheme are discretized using a fourth-order accurate method, and the four-step Runge–Kutta method provides fourth-order accuracy in time. Thus, the one-dimensional simulations are of fourth-order accuracy. In two-dimensional simulations, however, the advection fluxes are based on averaged values along the interface. Because the flux is not a linear function of the state quantities, fourth-order accuracy is downgraded to second-order accuracy. However, the use of a high-order interpolation scheme on the advection terms is able to ensure low diffusivity. This section conducts several simulations to validate the order of the accuracy of our numerical scheme.

1) 1D SIMULATION ERROR ANALYSIS

The full set of 1D compressible Euler equations permits acoustic waves. A 1D (x) domain of 3-km length with periodic boundary conditions is used to demonstrate the errors inherent in our scheme. The background pressure is set to 1×10^5 Pa, and in order to keep the acoustic speed at $c_s = 300\ m\ s^{-1}$, the background temperature is set to

$$T = \frac{c_s^2}{\gamma R_d}, \tag{70}$$

where γ is the adiabatic index, the ratio of specific heats of the gas at constant pressure to the gas at a constant volume (c_p/c_v), and R_d is the gas constant for dry air. The background density is calculated from

$$\rho = \frac{p}{R_d T}. \tag{71}$$

An initial temperature perturbation with a Gaussian distribution is added to the background temperature to initiate the acoustic waves:

$$T' = Ae^{-[(x-x_0)^2/s^2]}, \tag{72}$$

where $A = 0.5$ K, $x_0 = 1500$ m, and $s = 100$ m. The acoustic waves will travel in both directions at the speed of $c_s = 300\ m\ s^{-1}$ after the wave is fully developed. The result was sampled at $t = 48$ s to evaluate the order of accuracy of the numerical scheme. The grid spacing is varied from $dx = dz = 5$ to 40 m. The horizontal velocity at $t = 48$ s is shown in Fig. 6.

Since no analytical solution is available for this test, we treat the result with the finest resolution as the reference, and compare the results from coarser resolutions to this reference. To compare the results at different

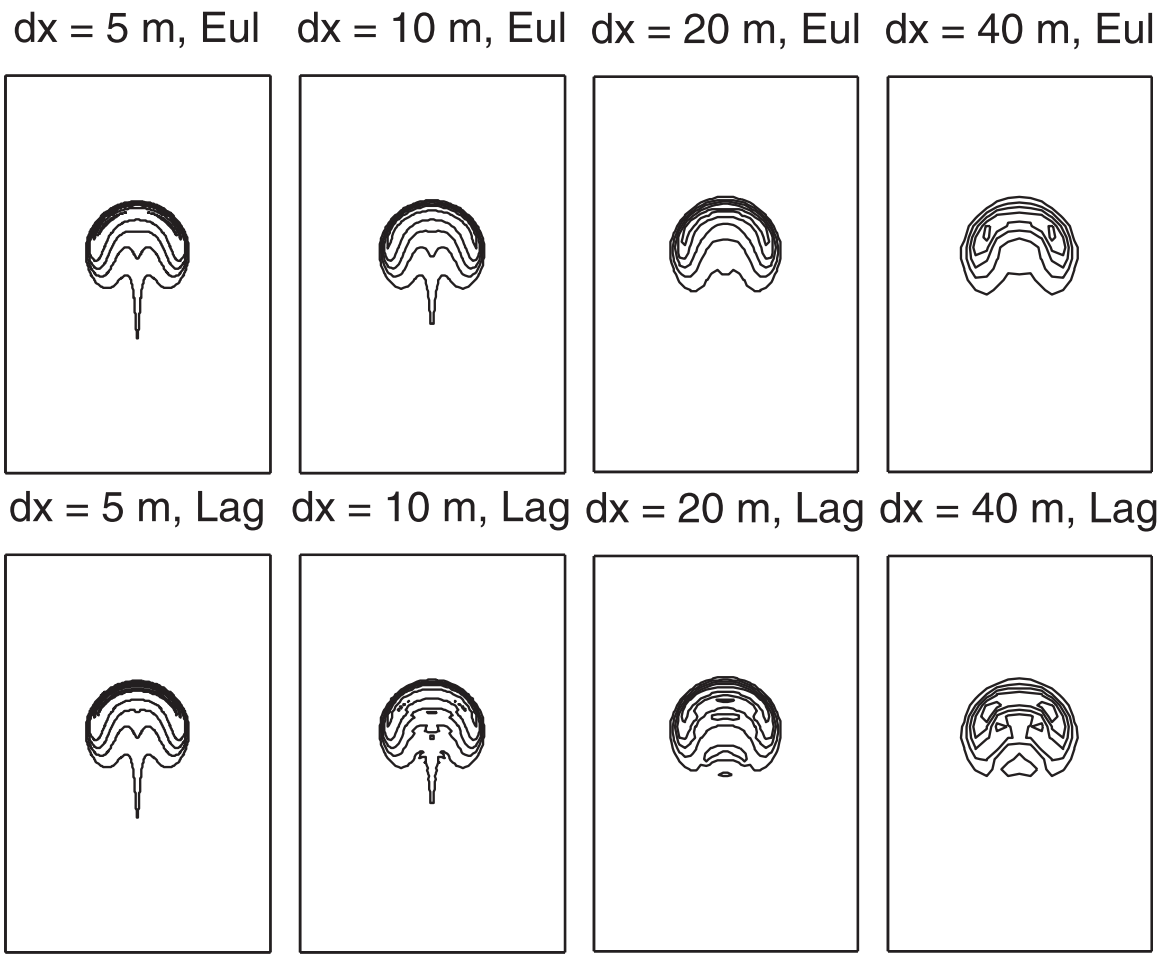

FIG. 5. The rising Gaussian bubble test for (from left to right) different $dx = dz$ resolutions simulated with (top) the Eulerian and (bottom) Lagrangian approach. A second-order explicit diffusion term with the coefficient set to 75 $m^2 s^{-1}$ is added to the system. The domain of the simulation is 1 km by 1.5 km (The vertical domain size is 1.52 km when the grid size of 40 m is used.) The contour interval is 0.1 K, and starts at 0.05 K.

resolutions, the reference solution was averaged to the coarser grids. Because finite-volume grids are used in our simulation, each grid point corresponds to the average value of a grid cell of width of dx, and for each coarsening step, since we doubled the grid spacing, we averaged the reference results to compare to these coarser resolutions (i.e., the result of cell size $dx = 10$ m was compared to the averaged value of two grid cells from the result with $dx = 5$ m, etc.). The averaged reference result is aligned with the results from coarser resolutions; the $l2$ norm error is calculated using the difference between the results from the coarser resolution and the averaged reference result. The logarithm plot of dx versus *error* is shown in Fig. 7a. The scattered points are in good agreement with the solid line, which has the slope of 4, so the one-dimensional test of our scheme is approximately fourth-order accurate.

2) ERROR ANALYSIS WITH 2D EULERIAN FORMULATION

We performed the rising bubble test using different resolutions. The Gaussian bubble tests with the Eulerian vertical coordinate were selected for the error analysis, because the profiles of the perturbation are more continuous. The sampled results are taken at a finite time of 6 min. Similar to the 1D simulation error analysis, the results from different resolutions using $dx = dz = 2.5$, 5, 10, and 20 m were used, and the result from $dx = dz = 2.5$ m were used as the reference result. Analogously to the one-dimensional test, the results from $dx = dz = 2.5$ m were averaged in order to compare them with results at different resolutions.

The $l2$ norm error is calculated using differences between the averaged reference result and results using

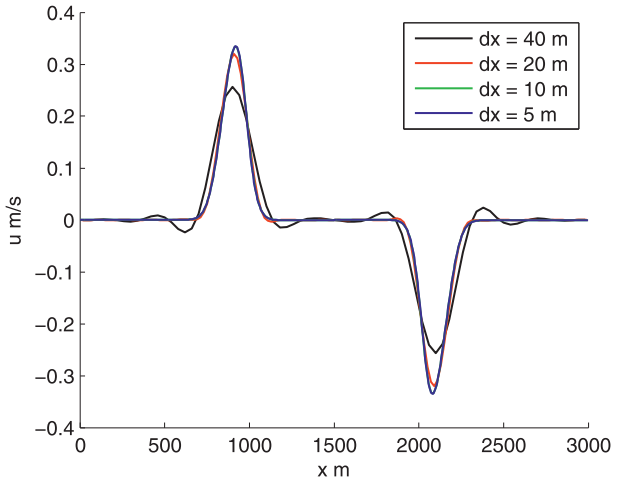

FIG. 6. Horizontal velocity at $t = 48$ s for the one-dimensional acoustic wave test at varying resolutions.

coarser resolutions. The logarithm plot of *dx* versus *error* is shown in Fig. 7b. The scattered points are in good agreement with the solid line, which has the slope of 2, so we conclude our two-dimensional test is approximately second-order accurate.

3) ERROR ANALYSIS WITH VERTICAL LAGRANGIAN COORDINATES AND REMAPPING

In the previous two sections we discussed the errors and order of accuracy of our model using a simple horizontal 1D and 2D Eulerian framework. In this section, the 2D Lagrangian framework together with remapping is examined. We used the gravity wave test described in section 4b of the paper. This analysis allows us to examine the accuracy of the Lagrangian framework for the transport of gravity waves. The gravity wave test with the Lagrangian vertical coordinate and a remap frequency of 60 s was selected for our error analysis. The sampled results are taken at a 3000 s. Similar to the warm bubble simulation error analysis, we used the result from the Lagrangian simulation at a resolution of $dx = dz = 250$ m as the reference. The reference result is averaged to compare to the Lagrangian results at resolutions of $dx = dz = 500$, 1000, and 2000 m.

The logarithm plot of *dx* versus *l*2 *error* is shown in Fig. 7c. The scattered points are in good agreement with the solid line, which has the slope of 2, so we conclude our two-dimensional tests using the vertical Lagrangian coordinate and remapping is approximately second-order accurate. Although the 1D simulation showed fourth-order accuracy, in 2D simulations, we did not use multiple points along the control-volume interfaces to describe the nonlinear distribution of the fluxes, so our scheme is downgraded to second-order accuracy. To reach a fourth order, one can use Gaussian quadrature points to calculate the interface fluxes (Ullrich et al. 2010).

c. *Comparison of the Eulerian and Lagrangian formulations without viscosity*

If the physical processes to be resolved do not involve the development of turbulence, the simulations of both Eulerian and Lagrangian formulations produce similar results. This is shown here by comparing the difference between early-time results from the rising bubble test

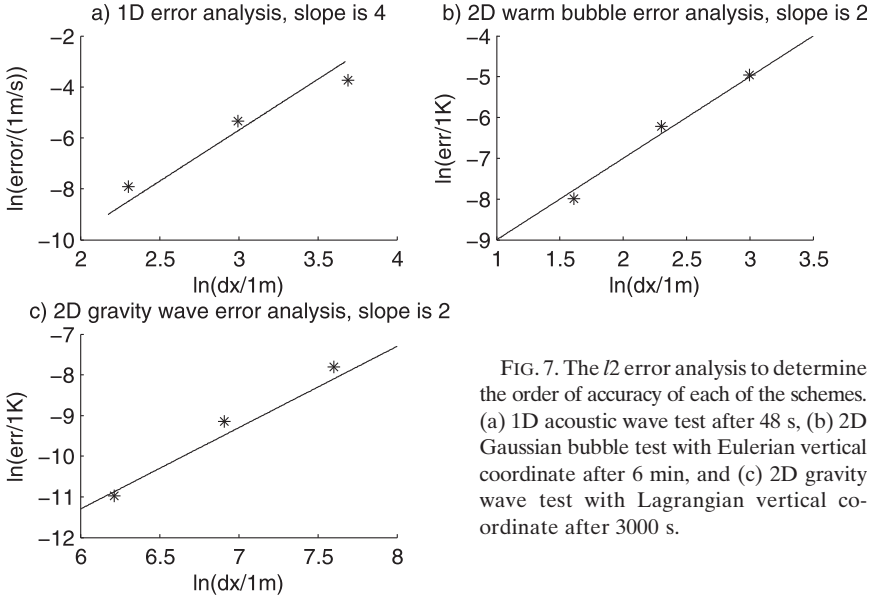

FIG. 7. The *l*2 error analysis to determine the order of accuracy of each of the schemes. (a) 1D acoustic wave test after 48 s, (b) 2D Gaussian bubble test with Eulerian vertical coordinate after 6 min, and (c) 2D gravity wave test with Lagrangian vertical coordinate after 3000 s.

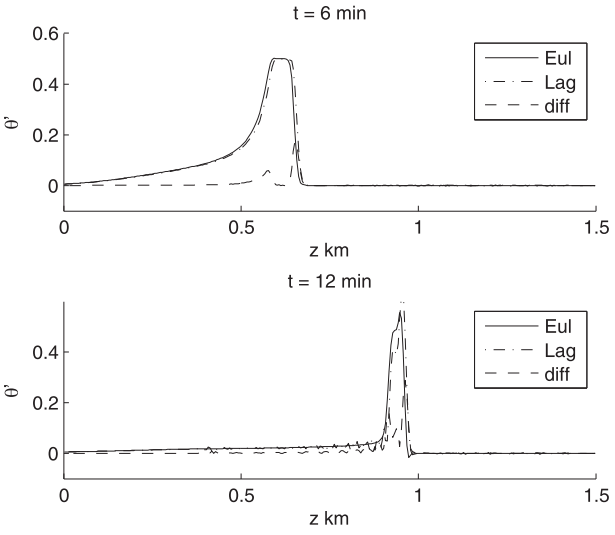

FIG. 8. Potential temperature perturbation in the Gaussian bubble test after (a) 6 and (b) 12 min sampled along the central vertical line at $x = 497.5$ m. In the legend, diff means the difference between the Eulerian and Lagrangian results (absolute value).

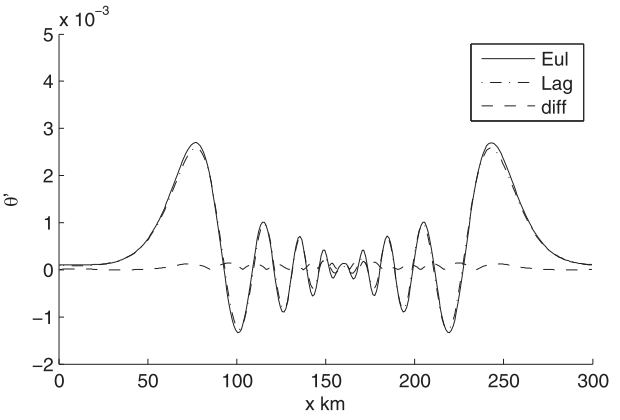

FIG. 9. Potential temperature perturbation in the gravity wave test after 3000 s sampled along the horizontal line at $x = 5.5$ km. In the legend, diff means the difference between the Eulerian and Lagrangian results (absolute value).

described in section 4a of the paper and the gravity wave test described in section 4b.

For the rising bubble test, we compare the plots of the Gaussian bubble at $t = 6$ and 12 min using a resolution of 5 m. The value of the potential temperature perturbation θ' is sampled along the central vertical line (i.e., at $x = 497.5$ m). The plot of θ'_{Eul}, θ'_{Lag}, and abs$(\theta'_{\text{Lag}} - \theta'_{\text{Eul}})$ is shown in Fig. 8.

The plot shows a slight shift between the Lagrangian bubble and the Eulerian bubble. However, the shape of the bubbles is almost identical as shown by the difference plot.

We conducted a similar examination of the gravity wave test, using a remap frequency of every 60 s for the Lagrangian formulation. The potential temperature is sampled along the central horizontal line (at $y = 5.5$ km) and at time $t = 3000$ s. The results are illustrated in Fig. 9.

Although the Lagrangian result is slightly more damped than the Eulerian result, the potential temperature perturbations calculated by both formulations are almost identical.

6. Conclusions

The finite-volume scheme using a vertical Lagrangian coordinate has proven to be very useful in the GCM modeling community and has been applied in both the CAM and GFDL GCMs. Here, we used a similar structure for the development of a nonhydrostatic dynamical core in 2D (x, z) Cartesian geometry.

In this work, we developed the equation sets for a generalized vertical coordinate, and present these in the nonhydrostatic Lagrangian and Eulerian form. We also show their hydrostatic variant. The Arakawa-A gridding is used in our approach to keep density, velocities and temperature all volume mean variables. With the A grid, the fluxes between the FVs or the vertical movement of the Lagrangian layer interfaces are calculated by Riemann solvers.

The Low Mach number Approximate Riemann Solver is designed for atmospheric fluid motions, and is extremely efficient when compared with the traditional (approximate/exact) Riemann solvers. With the introduction of LMARS, the numerical treatment of the fluxes is decoupled from the governing equations. The algorithm developer can chose variables or equations based on the specific physical requirement without changes to the numerical properties of the system.

No limiters or divergence damping are included in the numerical algorithm. The treatment for the interpolation and remapping are all based on polynomial interpolation. All plots present the ''pure'' effect of LMARS, and the two-grid-size wave does not grow or cause any instability. In real applications however, limiters are desirable to prevent negative values of density, pressure or tracers, like water vapor mixing ratios. Also, a better remap scheme will be required when simulating physical phenomena at the boundaries of the domain. But both limiters and remapping schemes cause extra diffusion in the numerical system, which, however, do not introduce any instability properties. In future studies, we will introduce limiters and an improved remapping scheme into our system.

LMARS is a flexible way to ensure stability for finite-volume numerical schemes both in Eulerian and

Lagrangian configurations. Of course for the simulations of clouds in a model, some form of explicit diffusion needs to be added to parameterize the subgrid-scale turbulence. However, with the built-in stabilizing mechanism in LMARS, there is no need to add an explicit diffusion term to the numerical scheme in for stability purpose. The numerical diffusion of the scheme can be decreased by using a high-order interpolation scheme or a smaller grid size or by reducing the diffusion factor in the LMARS scheme Although our approach utilizes the A grid, the LMARS technique can also be used to provide the FV cell interface velocities and pressure for the C and D grids. As a result, it is possible to integrate this method into the Lin–Rood scheme, which is applied in many GCMs. However, taking this next step would also entail exploring vertical implicit methods, which we have not yet done. In the future, we will explore an application that couples the nonhydrostatic regime and algorithm to a hydrostatic regime using a vertical Lagrangian coordinate.

Acknowledgments. The authors thank the two anonymous reviewers and the editor for their very helpful commentary in improving this work. This work was supported by the Office of Science, U.S. Department of Energy, Award DE-FG02-01ER63248.

REFERENCES

Arakawa, A., and V. R. Lamb, 1977: Computational design of the basic dynamical processes of the UCLA general circulation model. *General Circulation Models of the Atmosphere,* J. Chang, Ed., Vol. 17, *Methods in Computational Physics,* Academic Press, 173–265.

Bates, J., S. Moorthi, and R. Higgins, 1993: A global multilevel atmospheric model using a vector semi-Lagrangian finite-difference scheme. Part I: Adiabatic formulation. *Mon. Wea. Rev.,* **121,** 244–244.

Daley, R., 1988: The normal modes of the spherical non-hydrostatic equations with applications to the filtering of acoustic modes. *Tellus,* **40A,** 96–106.

Donner, L. J., and Coauthors, 2011: The dynamical core, physical parameterizations, and basic simulation characteristics of the atmospheric component AM3 of the GFDL Global Coupled Model CM3. *J. Climate,* **24,** 3484–3519.

Durran, D. R., 2010: *Numerical Methods for Fluid Dynamics: With Applications to Geophysics.* 2nd ed. Springer, 532 pp.

Giraldo, F. X., and M. Restelli, 2008: A study of spectral element and discontinuous Galerkin methods for the Navier–Stokes equations in nonhydrostatic mesoscale atmospheric modeling: Equation sets and test cases. *J. Comput. Phys.,* **227,** 3849–3877, doi:10.1016/j.jcp.2007.12.009. [Available online at http://dl.acm.org/citation.cfm?id=1349879.1349936.]

Hirsh, C., 2007: *Numerical Computation of Internal and External Flows: Fundamentals of Computational Fluid Dynamics.* 2nd ed. Butterworth-Heinemann, 680 pp.

Jablonowski, C., M. Herzog, J. E. Penner, R. C. Oehmke, Q. F. Stout, B. van Leer, and K. G. Powell, 2006: Block-structured adaptive grids on the sphere: Advection experiments. *Mon. Wea. Rev.,* **134,** 3691–3713.

Juang, H., 1992: A spectral fully compressible nonhydrostatic mesoscale model in hydrostatic sigma coordinates: Formulation and preliminary results. *Meteor. Atmos. Phys.,* **50,** 75–88.

Kasahara, A., 1974: Various vertical coordinate systems used for numerical weather prediction. *Mon. Wea. Rev.,* **102,** 509–522.

Klemp, J. B., W. C. Skamarock, and J. Dudhia, 2007: Conservative split-explicit time integration methods for the compressible non-hydrostatic equations. *Mon. Wea. Rev.,* **135,** 2897–2913.

Laprise, R., 1992: The Euler equations of motion with hydrostatic pressure as an independent variable. *Mon. Wea. Rev.,* **120,** 197–207.

Lin, S.-J., 2004: A ''vertically Lagrangian'' finite-volume dynamical core for global models. *Mon. Wea. Rev.,* **132,** 2293–2307.

——, and R. B. Rood, 1996: Multidimensional flux-form semi-Lagrangian transport schemes. *Mon. Wea. Rev.,* **124,** 2046–2070.

——, and ——, 1997: An explicit flux-form semi-Lagrangian shallow-water model on the sphere. *Quart. J. Roy. Meteor. Soc.,* **123,** 2477–2498.

Liou, M.-S., 2006: A sequel to AUSM, Part II: AUSM$^+$-up for all speeds. *J. Comput. Phys.,* **214,** 137–170.

Miller, M. J., and R. P. Pearce, 1974: A three-dimensional primitive equation model of cumulonimbus convection. *Quart. J. Roy. Meteor. Soc.,* **100,** 133–154.

——, and A. A. White, 1984: On the non-hydrostatic equations in pressure and sigma coordinates. *Quart. J. Roy. Meteor. Soc.,* **110,** 515–533.

Neale, R. B., and Coauthors, 2010: Description of the NCAR Community Atmosphere Model (CAM 5.0). NCAR Tech. Note NCAR/TN-486+STR, 268 pp. [Available online at http://www.cesm.ucar.edu/models/cesm1.1/cam/docs/description/cam5_desc.pdf.]

Phillips, N. A., 1956: The general circulation of the atmosphere: A numerical experiment. *Quart. J. Roy. Meteor. Soc.,* **82,** 123–164.

Putman, W. M., and S.-J. Lin, 2008: A finite-volume dynamical core on the cubed-sphere grid. *Proc. Astronomical Society of the Pacific Conf. Series Numerical Modeling of Space Plasma Flows: Astronum-2008,* Vol. 406, St. John, U.S. Virgin Islands, Astronomical Society of the Pacific, 268–276.

Robert, A., 1993: Bubble convection experiments with a semi-implicit formulation of the Euler equations. *J. Atmos. Sci.,* **50,** 1865–1865.

Skamarock, W. C., 2008: A linear analysis of the NCAR CCSM finite-volume dynamical core. *Mon. Wea. Rev.,* **136,** 2112–2119.

——, and J. B. Klemp, 1992: The stability of time-split numerical methods for the hydrostatic and non-hydrostatic elastic equations. *Mon. Wea. Rev.,* **120,** 2109–2127.

——, and ——, 1994: Efficiency and accuracy of the Klemp-Wilhelmson time-splitting technique. *Mon. Wea. Rev.,* **122,** 2623–2630.

——, and ——, 2008: A time-split nonhydrostatic atmospheric model for weather research and forecasting applications. *J. Comput. Phys.,* **227,** 3465–3485, doi:10.1016/j.jcp.2007.01.037. [Available online at http://portal.acm.org/citation.cfm?id=1347465.1347775.]

Smagorinsky, J., 1963: General circulation experiments with the primitive equations: I. The basic experiment. *Mon. Wea. Rev.,* **91,** 99–164.

Smolarkiewicz, P. K., L. G. Margolin, and A. A. Wyszogrodzki, 2001: A class of non-hydrostatic global models. *J. Atmos. Sci.,* **58,** 349–364.

Ullrich, P. A., C. Jablonowski, and B. van Leer, 2010: Riemann-solver-based high-order finite-volume models for the shallow-water equations on the sphere. *J. Comput. Phys.,* **229,** 6104–6134.

Whitehead, J., C. Jablonowski, R. B. Rood, and P. H. Lauritzen, 2011: A stability analysis of divergence damping on a latitude–longitude grid. *Mon. Wea. Rev.,* **139,** 2976–2993.

Xue, M., and A. J. Thorpe, 1991: A mesoscale numerical model using the non-hydrostatic pressure-based coordinate equations: Model experiments with dry mountain flows. *Mon. Wea. Rev.,* **119,** 1168–1185.

Yang, D., 2001: *C++ and Object-Oriented Numeric Computing for Scientists and Engineers.* Springer, 464 pp.

Yeh, K., J. Côté, S. Gravel, A. Méthot, A. Patoine, M. Roch, and A. Staniforth, 2002: The CMC-MRB Global Environmental Multiscale (GEM) model. Part III: Nonhydrostatic formulation. *Mon. Wea. Rev.,* **130,** 339–356.