Available online at www.sciencedirect.com
SCIENCE DIRECT®
Journal of Computational Physics 190 (2003) 601–622
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

An all-scale anelastic model for geophysical flows: dynamic grid deformation

Joseph M. Prusa [a,*], Piotr K. Smolarkiewicz [b]

[a] *Iowa State University, Ames, IA 50010, USA*
[b] *National Center for Atmospheric Research, Boulder, CO 80307, USA*

Received 9 July 2002; received in revised form 25 February 2003; accepted 19 May 2003

Abstract

We have developed an adaptive grid-refinement approach for simulating geophysical flows on scales from micro to planetary. Our model is nonoscillatory forward-in-time (NFT), nonhydrostatic, and anelastic. The major focus in this effort to date has been the design of a generalized mathematical framework for the implementation of deformable coordinates and its efficient numerical coding in a generic Eulerian/semi-Lagrangian NFT format. The key prerequisite of the adaptive grid is a time-dependent coordinate transformation, implemented rigorously throughout the governing equations of the model. The transformation enables mesh refinement indirectly via dynamic change of the metric coefficients, while retaining advantages of Cartesian mesh calculations (speed, low memory requirements, and accuracy) conducted fully in the computational domain. Diverse test results presented in this paper – simulations of a traveling stratospheric inertio-gravity-wave packet (with numerically advected dense-mesh region) and an idealized climate of the Earth (with analytically prescribed adaptive transformations) – demonstrate the potential and the efficacy of the new deformable grid model for tracing targeted flow features and dynamically adjusting to prescribed undulations of model boundaries.

1. Introduction

Astonishing progress in numerical methods and computer hardware during the past two decades have finally made realistic (3D transient) simulations of atmospheres and oceans feasible. However, inadequacies of boundary and initial conditions, grid resolution, and subgrid-scale physical parameterizations, still limit the accuracy obtainable with the available resources. A typical example is that of tropical storm forecasting: although steady progress has been made in the past 10 years in forecasting storm tracks (primarily due to improved data assimilation, [2]) other statistics, such as storm intensity, structure, and size, remain poorly predicted. The latest high-resolution hurricane simulations hint that it may be necessary to implement

* Corresponding author.
E-mail addresses: prusa@iastate.edu (J.M. Prusa), smolar@ncar.ucar.edu (P.K. Smolarkiewicz).

0021-9991/$ - see front matter © 2003 Elsevier Science B.V. All rights reserved.
doi:10.1016/S0021-9991(03)00299-7

global models [1] with mesoscale resolution, in order to significantly improve upon current tropical storm forecasts beyond one to two days. Global models with resolutions $\sim$10 km – a scale that just barely begins to resolve structure like the hurricane eye and rainbands – are still beyond the state of the art (several times in terms of the computational effort) using advanced solvers and massively parallel computer technologies. Even higher resolutions of $\sim$1 km – necessary to fully resolve these features – would require another $\sim$1000-fold increase in computational power.

In this paper, we outline a generalization of a coordinate transformation technique whereby the governing equations for the problem at hand are mapped into a transformed, or computational domain $\mathbf{S}_t$, which is both regular and stationary. Obvious advantages over solving the problem in the original physical domain, $\mathbf{S}_p$, are that: (i) irregular and time variable boundaries may be rigorously accommodated, and (ii) grid clustering in selected regions of interest may be naturally implemented. [2]

Analytical coordinate transformations have been used for over a century in continuous media, to simplify domains or extend known solutions. The explicit use of similarity transformations to solve phase change problems date back at least to Boltzmann in 1894 [3]; and the use of conformal mappings to solve problems in heat and potential fluid flow were on a solid basis by the end of the 19th century. In 1938, Prandtl [28] used an elementary translation of the boundary layer coordinate to transform the flat-plate solution into one along a curved plate (known as the *transposition theorem*). In 1949, Landau [21] used a simple algebraic transformation to translate and normalize a coordinate for predicting ablation rates. Numerous other studies appeared at this time and in subsequent years – and a large body of work now exists that has advanced the use of analytically specified transformations (e.g., either algebraic or conformal mappings).

A 1974 paper by Thompson et al. [46] is generally considered to be the seminal work on the *numerical generation* of curvilinear coordinates. In [46], the coordinates were computed by solving a posited set of elliptic equations. Brackbill and Saltzman [5] also generated coordinates by solving elliptic equations but derived formally as Euler–Lagrange equations that result from the extremization of a "mesh-adaptivity" functional. The significance of this optimization was that the mesh-generator could be constructed by taking into account properties of the grid (e.g., orthogonality and smoothness) and of the solutions (e.g., gradients of dependent variables) to minimize the truncation error. However, time-continuity is not readily accounted for (in the mesh-adaptivity functional) [3] so grid generation via a variational optimization appears best-suited for steady flows. In contrast, our concerns are with inherently transient problems, so we focus on time-continuous grid deformation.

By the early eighties, the use of continuous mappings, based upon either analytical or numerical methods, was recognized as continuous dynamic grid adaptation (CDGA), and became widespread in computational engineering. Meteorological applications, however, have favored the use of nested grids [8,32,34] over traditional CDGA methods. [4] Nested grids offer simplicity in algorithmic construction, and an ability to add or remove grid points as desired. In spite of these advantages, the abrupt changes in grid resolution near boundaries makes the development of satisfactory boundary conditions difficult, whereupon nested-grid methods are claimed unsuitable for long time integrations; cf. [14]. Early meteorological uses of CDGA include the works of Gal-Chen and Sommerville [16] on the analytic terrain-following

[1] Genesis and/or sudden changes in intensity of tropical storms appears to be "hyper" sensitive to lateral boundary conditions, whereupon simulations on the entire globe obviate the problem.

[2] In applications addressed in this paper, neither $\mathbf{S}_p$ or $\mathbf{S}_t$ are curved in the sense of geometrodynamics, with a nonzero Riemann curvature tensor – our concern is with curvilinear coordinate descriptions of a flat (locally Lorentz) space; [26].

[3] Time adaptivity may enter the elliptic problem indirectly through either (i) time-dependent boundary conditions, or (ii) replacement of the elliptic system resulting from the variational approach with it's time derivative [1].

[4] A third type of dynamic grid adaptation is rapidly gaining favor – based upon the use of unstructured grids [4]; such techniques are well advanced in engineering applications [47].

coordinate transformation, and of Sharman et al. [31] on the numerical generation of terrain-fitted grids – each was concerned only with stationary grids. Later studies incorporating time-adaptivity include [11,13,19,29] – they addressed atmospheric applications through the range of scales from small, through meso, to regional. Only recently, stationary adaptive grids have been applied to global climate simulation [14,15]. In this paper we complement earlier works with the development of a formal mathematical-numerical framework that allows the use of CDGA in a large class of applications. Our exposition and implementation are tailored for anelastic nonhydrostatic models of atmospheres and oceans, through the range of scales from micro [17] to global [43]; yet the model can be applied in a much broader class of problems ranging from biomechanics [9] to solar physics [12]. We demonstrate – we believe for the first time – the use of CDGA with time variation in a global model.

Our model is introduced in Section 2.1 using a symbolic, vector representation that enhances physical interpretation. The more detailed description of the various terms that arise – needed for the computational algorithm – is then given in tensor form in Section 2.2. Both types of description have merit. The various terms that arise in the mathematical model (e.g., material derivative, gradient, and so on) have a tangible, physical existence that is independent of any coordinate-based description and is best understood *intuitively* using a symbolic, geometric representation. However, coordinate-based representations are necessary for computing the explicit form of all requisite terms. Since the precise form of the terms depends upon the coordinate system being used, a tensor representation is preferable. It allows the use of powerful theory to deduce formulae that are valid in all coordinate systems, i.e., *coordinate invariant forms*, while conveying the physical interpretation of the symbolic representation. An example of paramount importance to our model involves the velocity – three different forms are used. Working only with a symbolic representation, it is not obvious that the use of three distinct velocities is helpful for designing an efficient computational algorithm. A coordinate-based representation reveals this. However, the details depend upon the coordinate system. A tensor representation merges these velocity details into coordinate invariant forms.

We illustrate our theoretical developments with two distinct examples of mesoscale and planetary type flows. Taken together, they document the efficacy of the deformable grid model in diverse applications and, perhaps even more important, show several modes of utilization of the coordinate mapping. The latter illustrates well the potential of our approach for computational research in meteorology.

Our first example simulates a finite-amplitude, 2D inertio-gravity-wave packet, forming in a rotating mid-latitude environment characterized by a uniform wind and stratification. The deflection of the lower boundary is prescribed as a traveling disturbance in the form of a broad, shallow trough. Such a deflection may be thought as a rough 2D approximation of the tropopause (an internal free boundary between the lower atmosphere and the stratosphere) in an axially symmetric, low pressure cyclone. The trough is translated periodically in the zonal direction, driving rotating-stratified-fluid response through various flow regimes. Here, the (vertical) coordinate mapping adapts to the lower boundary in the spirit of the terrain-following transformation [16] extended to time-dependent lower boundaries [29]. Independently of the vertical coordinate, the horizontal coordinate is transformed so that a high-resolution "nested grid" follows the gravity-wave packet. The zonal coordinate transformation uses a top-hat profile for the mesh density, which is advected numerically using our shape-preserving NFT advection scheme. Aside from documenting substantial savings in computational expense (roughly a factor of two compared to the simulation using uniformly high resolution everywhere) this example shows how continuous coordinate mapping can be employed in lieu of grid nesting common in meteorological models [8,34].

Our second example is the Held–Suarez planetary benchmark flow that idealizes the Earth's weather and climate [18] using a stationary diabatic forcing. From a fluid dynamics viewpoint, the Held–Suarez problem represents thermally forced baroclinic instability on the sphere. In a sense, it bears striking resemblance to large-eddy-simulation studies of convective boundary layers [24], where simulated flows are both turbulent and stochastic. Although small differences in model setups can lead to totally different instantaneous flow realizations (viz. weather), the integral flow characteristics (viz. climate) are fairly stable and predictable.

We use evolution of such integral flow features (e.g., mid-latitude zonal jets) as a focus for the mesh adaptivity, and compare several different simulations with stationary (stretched/unstretched) and adaptive grids. Our results are dramatic in revealing potential of the mesh adaptivity for planetary flows. First, solutions based upon the stretched/adaptive grids show substantially more physical details (sharp yet smooth time-space contrasts) in the regions of interest (e.g., tropics) than the solutions employing uniform meshes of comparable computational complexity. Second – a feature unique to the cartography of the sphere – focusing grid resolution in equatorial regions, with its concomitant reduction near the poles, improves the conditioning of the numerical elliptic operator, thereby accelerating the convergence of the elliptic Krylov-subspace solver. Together, these two aspects of mesh stretching/adaptivity effect in delivering accuracy comparable to a uniform-grid result with twice the resolution for less than half the computational expense.

A few brief notes about notation are in order. We use $\sim$ to mean *on the order of*, $\equiv$ to mean *identically*, and $:=$ to mean *defined as*. The velocity vector $\mathbf{v}$ appearing in Section 2.1 does *not* distinguish between the three velocities mentioned above. In Section 2.2 these velocities *are* distinguished. Also, superscripts are of two distinct types, (i) repeating indices i, j, k, p, and q, and (ii) indices that denote the physical or geometric nature of a quantity, such as superscript s, *, and $'$. Finally, observe that indices may have ranges 1–3 or 0–3.

2. Anelastic fluid model

Because of the enormous span of the spatial and temporal scales important in geophysical fluids, explicit integrations of generic compressible equations are impractical (viz., prohibitively expensive) for most applications. As a result, meteorological models utilize a variety of analytic approximations to the fluid equations (hydrostatic, elastic, anelastic, Boussinesq, and so on) and evince many split-explicit or semi-implicit methods for their integrations. For research studies of all-scale geophysical fluids, we have found the anelastic nonhydrostatic system optimal so far.

2.1. Overview

For simplicity, here we focus the discussion on an inviscid, adiabatic, density-stratified fluid whose undisturbed, geostrophically balanced "ambient" (or "environmental") state is described by the potential temperature $\theta_e = \theta_e(\mathbf{x})$ and the velocity $\mathbf{v}_e = \mathbf{v}_e(\mathbf{x})$. To begin discussion, we present first a compact, symbolic form of the governing equations for the anelastic system of Lipps and Hemler [22]:

$$\mathbf{Div}(\rho_b \mathbf{v}) = 0, \tag{1}$$

$$\frac{\mathscr{D}\mathbf{v}}{\mathscr{D}t} = -\mathbf{Grad}(\pi') - \mathbf{g}\frac{\theta'}{\theta_b} + \mathbf{F}, \tag{2}$$

$$\frac{\mathscr{D}\theta'}{\mathscr{D}t} = -\mathbf{v} \bullet \mathbf{Grad}\theta_e, \tag{3}$$

Here the operators $\mathscr{D}/\mathscr{D}t$, **Grad**, and **Div** symbolize the material derivative, gradient, and divergence; $\mathbf{v}$ denotes the velocity vector; $\mathbf{F}$ symbolizes inertial forces (e.g., Coriolis and metric forces due to the curvature of the coordinates; see [41]); θ, ρ, and π denote potential temperature, density, and a density-normalized pressure; and $\mathbf{g}$ symbolizes the gravity vector. Primes denote deviations from the environmental state. The subscript $_b$ refers to the basic state, i.e., a horizontally homogeneous hydrostatic reference state of the Boussinesq expansion around a constant stability profile (see [8, Section 2b], for a discussion).

The anelastic Eqs. (1) and (2) may be viewed as combining two distinct approximations in the compressible Euler equations: a Boussinesq type linearization of the pressure gradient forces and mass fluxes in momentum and mass continuity equations, respectively; and the anelasticity per se equivalent to taking the limit of an infinite speed of sound. Both assumptions are well accepted in solar [12] and ocean [25] modeling. Although the anelastic equations have been proven accurate for simulating the elements of atmospheric weather and climate up to synoptic scales [27], their use raises concerns when extended to global circulations. Obviously, solving fully compressible Euler equations would circumvent such concerns. However, designing an accurate, flexible, and computationally efficient all-scale geophysical fluid model based on the fully compressible equations is a difficult task, especially in the class of NFT approximations advocated in [12,17,43,44] and pursued here. In contrast, adopting the anelastic approximation greatly simplifies the task. In [43], the authors discussed extensions of a Cartesian small- to mesoscale nonhydrostatic anelastic model to a rotating mountainous sphere. Using benchmark "dynamical core" experiments that idealize atmospheric weather and climate, they assessed the efficacy of numerous formulations of the global model by measuring the differences due to analytic formulation of the governing equations (hydrostatic, nonhydrostatic, compressible, anelastic, incompressible, etc.) against the truncation errors of optional second-order-accurate discretization approximations (implicit, explicit, semi-Lagrangian, Eulerian, etc.). They showed that the differences due to the higher-order truncation errors of legitimate modes of executing contemporary global models overwhelm the differences due to analytic formulation of the governing equations. Their study, and its extension to moist processes in [17], builds confidence that nonhydrostatic anelastic global models derived from small-scale codes (i.e., not relying on large-scale balances in their analytic/numerical design) adequately capture a broad range of planetary flows.

2.2. Analytic formulation

Flexibility to solve the governing equations in a variety of domains – a unique feature of our anelastic model – comes from a generalized coordinate transformation (homeomorphism) that maps the physical domain $\mathbf{S}_p$ onto a transformed, computational domain $\mathbf{S}_t$ while preserving the topology (connectivity) of $\mathbf{S}_p$. In this paper, the original physical domain $\mathbf{S}_p$ is described using either (i) Cartesian coordinates, or (ii) spherical coordinates with an $\mathcal{O}(\varepsilon)$ circle about the poles removed (where $\epsilon \sim$ meridional grid increment). [5] Thus, the same model can be used both for global and small-to-mesoscale applications. Embedded in the transformation is a time-variable version [29] of the standard terrain-following coordinates, and more significantly, a horizontal stretching whereby the horizontal coordinates in $\mathbf{S}_t$ are arbitrary (subject to C^2 continuity) functions of the time and horizontal coordinates in $\mathbf{S}_p$.

The three-dimensional, time variable mapping is specified as

$$(\overline{t},\overline{x},\overline{y},\overline{z}) \equiv (t, E(t,x,y), D(t,x,y), C(t,x,y,z)), \tag{4}$$

where in particular, $(\overline{x},\overline{y})$ do not depend upon the vertical coordinate z. The latter keeps vertical columns vertical, to preserve the primary hydrostatic structure of the atmosphere, and simplifies the metric terms.

In the transformed coordinates $(\overline{t},\overline{\mathbf{x}}) \in \mathbf{S}_t$, Eqs. (1)–(3) can be written as follows:

$$\frac{\partial(\rho^*\overline{v}^k)}{\partial\overline{x}^k} = 0, \tag{5}$$

[5] This replaces the traditional differentiation across the pole, used in [43], with Neuman boundaries on the $\mathcal{O}(\varepsilon)$ circle, thereby greatly improving communications in the massively parallel variant of the model code when grid deformation is extended to the vicinity of the poles.

$$\frac{\mathrm{d}v^j}{\mathrm{d}\overline{t}} = -\widetilde{G}^k_j \frac{\partial \pi'}{\partial \overline{x}^k} + g\frac{\theta'}{\theta_b}\delta^j_3 + F^j, \tag{6}$$

$$\frac{\mathrm{d}\theta'}{\mathrm{d}\overline{t}} = -\overline{v^s}^k \frac{\partial \theta_e}{\partial \overline{x}^k}, \tag{7}$$

where $\rho^* := \rho_b \overline{G}$, with $\overline{G}$ denoting the Jacobian of the transformation (defined below), and $j,k = 1,2,3$ correspond to the $\overline{x},\overline{y},\overline{z}$ components, respectively. In the momentum Eq. (6), $\widetilde{G}^k_j := \sqrt{g^{jj}}(\partial \overline{x}^k/\partial x^j)$ are renormalized elements of the Jacobi matrix where summation is not implied over j, and δ^j_3 is the Kronecker delta. The coefficients g^{jj} are the diagonal elements of the conjugate metric tensor of $\mathbf{S}_p$ (defined below).

Eqs. (5)–(7) need to be studied carefully – while they are reminiscent of the standard forms (1)–(3) as well as previously published tensor formulations [48], there are a few subtleties involved. The total derivative is given by $\mathrm{d}/\mathrm{d}\overline{t} = \partial/\partial\overline{t} + \overline{v^*}^k(\partial/\partial\overline{x}^k)$, where $\overline{v^*}^k := \mathrm{d}\overline{x}^k/\mathrm{d}\overline{t} := \dot{\overline{x}}^k$ is the *contravariant velocity*. Appearing in the continuity (5) and potential temperature (7) equations is the *solenoidal velocity*,

$$\overline{v^s}^k := \overline{v^*}^k - \frac{\partial \overline{x}^k}{\partial t} \tag{8}$$

so named because of the form continuity (5) takes with it. This form is a general result for the transformation (4) given: (i) $\rho_b = \rho_b(\mathbf{x})$; and (ii) the physical coordinate system $\mathbf{x} \in \mathbf{S}_p$ does not depend upon time (Cartesian, cylindrical, spherical, oblate spheroidal, and so on, would all be acceptable choices for the physical system). Eq. (5) readily follows from the tensor invariant form of anelastic continuity $\overline{G}^{-1}\partial(\rho_b \overline{G}\overline{v^*}^i)/\partial\overline{x}^i \equiv 0$ where $i = 0,1,2,3$ ($i = 0$ refers to time $\overline{t}$, thus $\overline{u^*}^0 \equiv 1$) [30]. Use of the solenoidal velocity has advantages for the solution procedures because it preserves the incompressible character of numerical equations. [6] In the momentum equation, it is the specified *physical velocity* v^j, defined in $\mathbf{S}_p$, that is advected with the total derivative. In meteorological applications, the physical velocity is typically defined using a local Cartesian system and so has dimensions of length/time. A distinct representation of the physical velocity, $\overline{v}^j \neq v^j$, also exists for the transformed coordinate system. If v^j is "typically defined", it is given by $\overline{v}^j = \sqrt{\overline{g}_{jj}}\ \overline{v^*}^j$, where summation is not implied. Then the physical $\overline{v}^j$ and contravariant $\overline{v^*}^j$ velocity differ by a scale factor only. The contravariant velocity is transformed according to $\overline{v^*}^j = (\partial\overline{x}^j/\partial x^i)v^{*^i}$. While these relationships suffice for expressing any velocity (solenoidal, contravariant, or physical) in terms of the others, a particularly useful relation for expressing the solenoidal velocity in terms of the physical is

$$\overline{v^s}^j = \widetilde{G}^j_k v^k. \tag{9}$$

The elements of the metric tensor of the transformed coordinates are $\overline{g}_{mn} = g_{pq}(\partial x^p/\partial\overline{x}^m)(\partial x^q/\partial\overline{x}^n)$, where g_{pq} denotes the metric tensor of the physical coordinate system (which need not be Cartesian). The Jacobian is then $\overline{G} = |\ \overline{g}_{mn}\ |^{1/2}$. The components of g_{pq} may be computed by identifying $\mathbf{S}_p$ as Riemannian with symmetric distance (or fundamental) metric $\mathrm{d}s^2 = g_{pq}\mathrm{d}x^p\mathrm{d}x^q$ [45]. This fundamental metric is readily computed using the Pythagorean Theorem on an infinitesimal element in any orthogonal coordinate system (e.g., $g_{pq} = \delta^p_q$ for Cartesian coordinates; while more generally $g_{pq} = 0$ for $p \neq q$ in orthogonal $\mathbf{S}_p$). Consequently, the components of the conjugate metric tensor, needed in (6), are computed from $g^{jj} = 1/g_{jj}$. [7] Note that unlike g^{pq}, the metric coefficients $\widetilde{G}^q_p$ appearing in Eqs. (6) and (9) are *not* symmetric (i.e., $\widetilde{G}^q_p \neq \widetilde{G}^p_q$).

[6] For moving coordinate systems, the anelastic continuity equation using contravariant velocities is not divergence free and takes, in effect, a compressible/elastic form [29].

[7] In nonorthogonal systems, g^{pq} may be evaluated from the relation $g_{pk}g^{kq} \equiv \delta^q_p$ [45].

Curvilinear coordinate descriptions generate apparent centrifugal and Coriolis accelerations simply due to the curvature of the coordinate system. These *Christoffel terms* [45] are placed into the forcing term, F^j, of Eq. (6). Geophysical Coriolis and centrifugal accelerations, as well as any additional forcings in the momentum equation due to frictional/stress terms such as damping (e.g., gravity-wave absorbers in vicinity of open boundaries) and/or subgrid-scale (SGS) modeling are also included in the F^j term. The magnitude of gravity is denoted by $g = |\mathbf{g}|$. Extensions to thermal forcing – such as radiation, damping and/or SGS modeling terms, can be incorporated into the model by adding appropriate terms to the right-hand side of (7).

2.3. Numerical approximations

Each prognostic equation of the anelastic system (6) and (7) can be written in a compact conservation-law form

$$\frac{\partial \rho^* \psi}{\partial \bar{t}} + \overline{\nabla} \bullet (\rho^* \bar{\mathbf{v}}^* \psi) = \rho^* R, \tag{10}$$

where $\overline{\nabla}\bullet := (\partial/\partial\bar{x}, \partial/\partial\bar{y}, \partial/\partial\bar{z})\bullet$ and ψ is an intensive dependent fluid variable, such as a component of specific momentum (viz. velocity component), or potential temperature. In (10), R combines all forcings and/or sources that appear on the right-hand side of (6) or (7). Given (5), (10) is mathematically equivalent to the Lagrangian evolution equation

$$\frac{\mathrm{d}\psi}{\mathrm{d}\bar{t}} = R. \tag{11}$$

Our basic NFT approach for approximating either (10) or (11) on a discrete mesh is second-order-accurate in space and time. The two optional model algorithms, Eulerian [36] and semi-Lagrangian [35], correspond to (10) and (11). Either algorithm can be written in the compact form

$$\psi_{\mathbf{i}}^{n+1} = \mathrm{LE}_{\mathbf{i}}(\widetilde{\psi}) + 0.5\Delta t R_{\mathbf{i}}^{n+1}. \tag{12}$$

Here, we denote $\psi_{\mathbf{i}}^{n+1}$ as the solution at the grid point $(\bar{t}^{n+1}, \bar{\mathbf{x}}_{\mathbf{i}})$; $\widetilde{\psi} := \psi^n + 0.5\Delta t R^n$; and *LE* denotes either an advective semi-Lagrangian or a flux-form Eulerian NFT transport operator. In the Eulerian scheme, LE integrates the homogeneous transport Eq. (10), i.e., LE advects $\widetilde{\psi}$ using a fully second-order-accurate multidimensional NFT advection scheme [40,44]. In the semi-Lagrangian algorithm, LE remaps transported fields, which arrive at the grid points $(\bar{t}, \bar{\mathbf{x}}_{\mathbf{i}})$, back to the departure points of the flow trajectories $(\bar{t}^n, \bar{\mathbf{x}}_0(\bar{t}^{n+1}, \bar{\mathbf{x}}_{\mathbf{i}}))$ also using NFT advection schemes [34,35].

Eq. (12) represents a system of equations that is implicit with respect to all dependent variables in (6) and (7), since all forcing terms are assumed to be unknown at $n+1$. Note that the forcing term on the rhs of (7) contains the complete convective derivative. This is significant because it guarantees that the implicitness of the numerical approximation does not adversely affect either the impermeability of the lower boundary or the conservation of θ', regardless of details of the transformation (4) (see [10,43], for discussions). The implicitness of the pressure gradient forces is an essential feature of the anelastic model, as it enables the projection of the preliminary values $\mathrm{LE}(\widetilde{\psi})$ onto solutions of the continuity Eq. (5). To make this projection, the system of simultaneous equations resulting from (12) are algebraically inverted to construct expressions for the solenoidal velocity components. These components are substituted into (5) to produce an elliptic equation for pressure (see Appendix A for the complete development). The elliptic pressure equation is solved (subject to appropriate boundary conditions) using the generalized conjugate-residual approach – a preconditioned nonsymmetric Krylov-subspace solver [33,37,38]. Given the updated pressure, and hence the updated solenoidal velocity, the updated physical and contravariant velocity components are

constructed from the solenoidal velocities using Eqs. (9) and (8), respectively. Nonlinear terms in R^{n+1} (e.g., metric terms arising on the globe) may require outer iteration of the system of equations generated by (12) – see [43, Appendix] for discussion. When included, subgrid-scale modeling is typically first-order-accurate and explicit, i.e., it assumes $\mathrm{SGS}(\psi^{n+1}) = \mathrm{SGS}(\psi^n) + \mathcal{O}(\Delta t)$ in R^{n+1}, see in [40, Section 3.5.4]. For extensions to moist processes, see [17].

3. Example simulations using time-adaptive grids

The CDGA (continuous dynamic grid adaptation) capability is enabled by moving the physical grid in $\mathbf{S}_t$. The following subsections document this capability with two complementary examples. The first example demonstrates that the mathematical formalism and NFT algorithms that underlie the model allow flexibility in effecting coordinate transformations. In this case, we numerically advect an interior nested grid of enhanced uniform resolution in order to follow the development of a two-dimensional inertio-gravity-wave packet. The nested grid is embedded in an exterior grid of reduced uniform resolution. Consistent with our NFT model algorithm, no apparent numerical artifacts (e.g., dispersive oscillations) are observed regardless of the step changes in grid resolution. The second example demonstrates the time adapted grid capability in a far larger application – a fully three-dimensional, nonhydrostatic, global atmospheric model. Here the adaptation is analytically specified to trace the development of mid-latitude, zonal jets. The flow is turbulent and stochastic – so we present a few selected statistics on the "climate" to gauge the effectiveness of the adaptive grid.

3.1. Traveling boundary disturbance

This experiment simulated the evolution of an inertio-gravity-wave packet generated by a 2D depression traveling along the tropopause using a mesoscale sized, Cartesian physical domain. We used a cosine deflection of the lower domain boundary (streamline) $z_s(t,x) = -0.5h_0[1+\cos(\pi x'/L)]$ if $|x'|/L \leqslant 1$ and $z_s(t,x) = 0$ otherwise, with $x' := x - (x_0 + \mathscr{L}\sin(2\pi t/T))$. The depth and half-width of the deflection were assumed $h_0 = 500$ m, and $L = 200$ km, respectively. The amplitude of the deflection's horizontal displacement and the period of oscillation were set to $\mathscr{L} = 3000$ km and $T = 96$ h, respectively. The basic state (the environmental state was set identically) was one of uniform zonal wind ($v_b = 10$ m s^{-1}), stability (with Brunt–Väisälä frequency $N = 0.02$ s^{-1}), infinite density scale height (i.e., Boussinesq approximation), and Coriolis parameter set for +45° latitude.

The transformed vertical coordinate was prescribed as the time variable generalization [29] of terrain following coordinates [16] – $\overline{z} \equiv C(t,x,y,z) = H(z-z_s)/(H-z_s)$, where H is the depth of the model domain. The reference computational grid consisted of 1001×201 points, and covered the model domain of $[-2500, 2500] \times [0, 40]$ km^2 with a uniform resolution of $\Delta\overline{x} = 5.0$ km and $\Delta\overline{z} = 0.2$ km in the zonal and vertical, respectively. The simulation spawned half of the period of the lower boundary oscillation with $NT = 5760$ uniform intervals $\Delta t = 30$ s. The open boundaries of the model used weak sponges to minimize spurious wave reflections. At $t = 0$, the initial condition consisted of the lower boundary deflection centered at $x_0 = -1500$ km, flat isentropes, and the potential flow satisfying the incompressibility constraint (5) and boundary conditions. The evolution of the reference solution is illustrated in Fig. 1 that displays isolines of the vertical velocity field every 8 h. Note that the magnitude of the flow and the dominant vertical wavelength vary in time.

The result in Fig. 1 serves as the reference for an extreme utilization of the adaptive-grid formalism presented in Section 2.2. Here, we employ the continuous transformation (4) in the limit where the transition between two homogeneous grids with distinct horizontal resolutions occurs in a distance smaller than the horizontal grid increment itself. The coordinate transformation is computed numerically – in the spirit

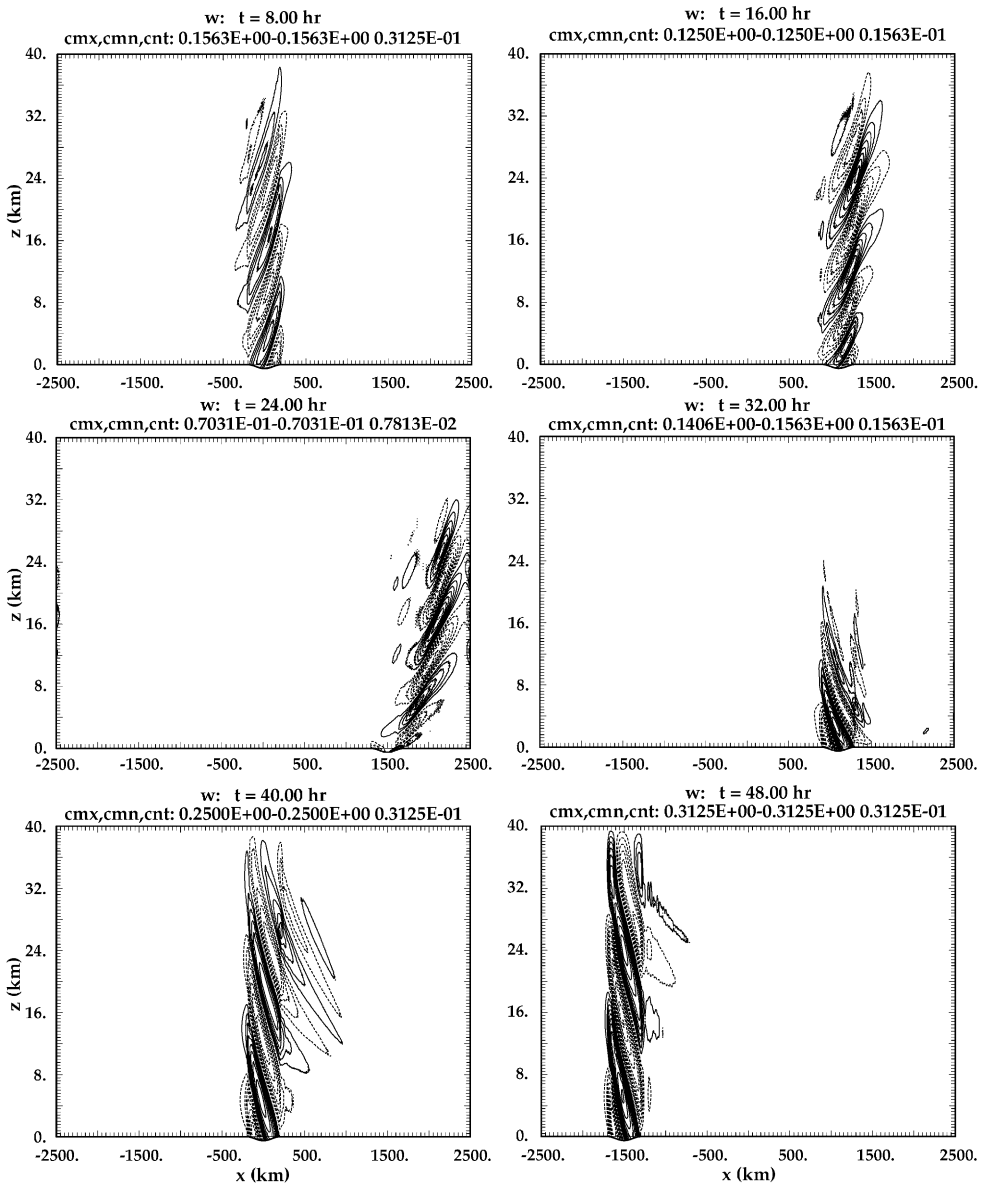

Fig. 1. Traveling inertio-gravity-wave packet; reference solution generated using a uniform, high-resolution grid.

of using advection schemes derived via continuous arguments of Taylor-series expansion for transporting fronts and flow discontinuities. From a model-engineering viewpoint, this example documents how continuous coordinate mappings can be implemented to mimic nested grids. All setups of the CDGA simulation are the same as in the reference runs, except that the horizontal grid is twice coarser outside the region of the traveling wave packet $(x_0 + \mathscr{L}\sin(2\pi t/T)) \pm 500$ km (thus the grid consisted of 601×201 points).

The movement of the nested grid can be specified quite arbitrarily with disregard to details of the simulated flow – for its utility requires that it encompasses the evolution of physical features of interest only

"reasonably close". Forthcoming from such a principle, we define the initial physical grid (here Cartesian) with an embedded dense nest essentially "by hand", and then postulate its evolution via a "mesh-continuity" conservative advection equation posed entirely in the transformed domain

$$\frac{\partial \delta_x}{\partial \bar{t}} + \frac{\partial U \delta_x}{\partial \bar{x}} = 0. \tag{13}$$

Here $\delta_x = \delta_x(\bar{t}, \bar{x})$ denotes the physical-grid-increment function of the transformed coordinates, [8] and U is the velocity of the mesh-movement. With (13), the entire problem of prescribing the mesh movement has been placed in the definition of U. In general, Eq. (13) will have a δ_y counterpart, and each of the two mesh-continuity equations will include divergence of the advective fluxes ($V\delta_x$ and $V\delta_y$, respectively) in $\bar{y}$ direction. To maintain uniformity of both dense and coarse meshes, U and V need to satisfy certain conditions – zero divergence and limited deformation, at least. In the present example, we select U equal to the velocity of the traveling depression in transformed coordinates, i.e., $U = U(\bar{t}) = \beta 2\pi \mathscr{L}/T \cos(2\pi t/T)$, with $\beta = 5/6$ (see Appendix B for details).

Eq. (13) is solved numerically at each time step of the model – prior to updating $R_{\mathbf{i}}^{n+1}$ forcings in (12) – using a monotone, flux-form NFT advection scheme. The conservative formulation is important to assure preservation of the total length of the domain. Having updated $\delta_x|^{n+1}$, new values of the $x(\bar{t}, \bar{x})$ coordinates are found by averaging (to prevent a round-off error build-up) the two recurrences: $x_{i+1} = x_i + \delta_x|_{i+1/2}$ and $x_{i-1} = x_i - \delta_x|_{i-1/2}$, with x_1 and x_{NX} held constant at the left and right boundaries of the model domain. Note that δ_x does not need to be stored, but it can be defined from $\delta_x|^n_{i+1/2} := x^n_{i+1} - x^n_i$ before the call to the advection routine integrating (13).

Fig. 2 shows the adaptive solution at four selected times to compare with the reference solution in Fig. 1. The present results are plotted in the transformed domain so that no smoothing due to interpolation is introduced. As a result, the proportions of the two figures differ, but they show excellent comparability, down to fine details. For the sake of completeness, the geometry of the mesh is illustrated in Fig. 3. Solid lines show computational results for normalized mesh density $\delta_x \Delta\bar{x}^{-1}$ whereas dashed lines show physical coordinate x normalized by the size of the domain. [9] The abscissa is the transformed coordinate $\bar{x}$ normalized by the domain size. An exact solution (see Appendix B) shows that $\delta_x \Delta\bar{x}^{-1} = 6/5$, 3/5 for the coarser and finer grids, respectively, and that the horizontal extent of the finer grid in normalized, transformed coordinates is 1/3. The numerical solution depicted in Fig. 3 agrees with these values precisely.

The mesh-continuity equation has been integrated using the third-order-accurate, fully monotone option of the multidimensional positive definite advection transport algorithm (MPDATA) [23]. In spite of the considerable expense of this scheme and the overhead associated with the CDGA machinery, the adaptive solution requires only 60% of the computational expense of the reference solution for essentially the same accuracy.

3.2. Idealized climate simulations

This second example simulates the idealized climates of Held and Suarez [18], and demonstrates the potential of mesh adaptivity for modeling geophysical turbulence. The Held–Suarez test problem typifies the response of an initially stagnant and uniformly stratified fluid to a diabatic forcing that mimics the long-term thermal and frictional forcing in the Earth's atmosphere. This diabatic forcing attenuates θ and $\mathbf{v}$ to the specified equilibrium temperature $\theta_{EQ}(|y|, z)$ and $\mathbf{v}|_{z<z_i} = 0$, where z_i represents a height of the boundary layer (see [18, Section 2], for details); y and z refer to the meridional and the vertical (radial) directions,

[8] Since the transformed grid is always homogeneous, δ_x is proportional to a finite difference representation of a grid stretching factor $\partial x/\partial \bar{x}$.

[9] The domain size is selected the same for both the transformed and physical domains.

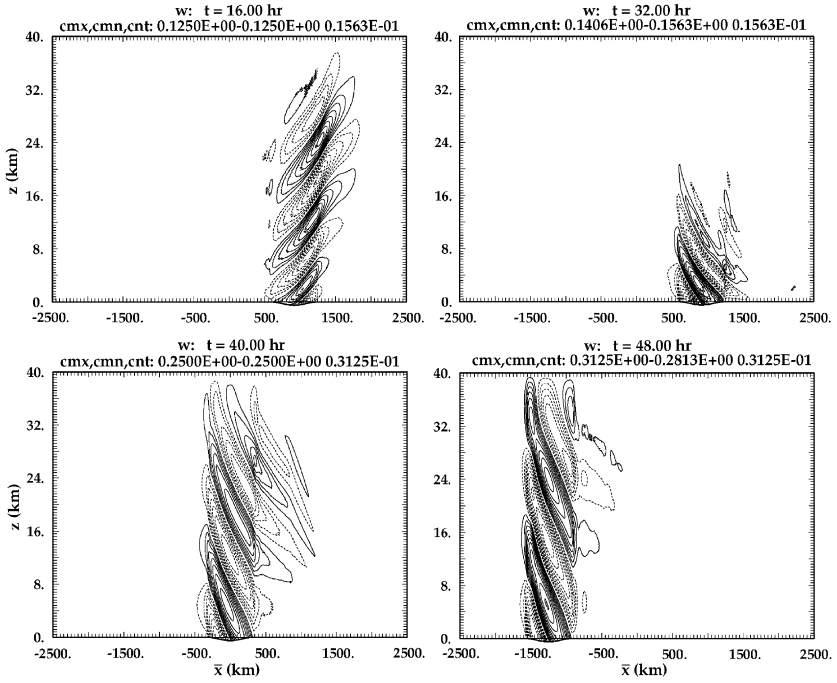

Fig. 2. Traveling inertio-gravity-wave packet; CDGA solution with 1000 km region of high grid resolution only in immediate neighborhood of the traveling disturbance.

respectively. The corresponding forcing functions augment the governing equations of motion (6), (7) with appropriate Rayleigh friction and Newtonian cooling/heating terms.

Results from four numerical experiments, that differ only in the design of the horizontal grid, are summarized in Table 1, and Figs. 4 and 5. U0 and U2 refer to stationary, uniform-increment grids in the zonal and meridional directions. These two simulations provide our "coarse" and "fine" meridional-resolution control results. Grid SS is also stationary but with doubled uniform zonal resolution. In the meridional direction, a coordinate transformation is applied such that a broad equatorial region has double the resolution of the uniform grid U0. Consequently, we regard the SS simulation to be the most accurate in the equatorial region. Grid TS is time adaptive, initially with uniform zonal and meridional increments. At 50 days the meridional coordinates begin to adapt to the developing zonal structure, such that at 150 days the region near $\pm 37^\circ$ latitude has double the resolution of a uniform grid, while a narrow equatorial band maintains negligible change in grid resolution. This puts the maximum meridional resolution approximately in the region of the mid-latitude zonal jets. Outside of the time interval 50–150 days, the TS grid is stationary. The time-dependent coordinate transformation underlying the grid adaptation (as well as the static stretch for case SS) was set up a priori in the form of elementary polynomial functions, discussed in [29]. [10] Each of the four simulations was run for 3 years, beginning with a randomly perturbed no-flow initial state, see [43] for further details of the U0 simulation. Finally we note that no explicit subgrid-scale (SGS) turbulence model was used in this simulation – in spite of the strong turbulence evinced by the flow. Instead, we relied upon the nonoscillatory machinery of the NFT algorithms to provide the dissipation

[10] Recall, from the preceding example, that our NFT model is equally adept at handling the time-dependent coordinate transformation numerically in response to developing flow features.

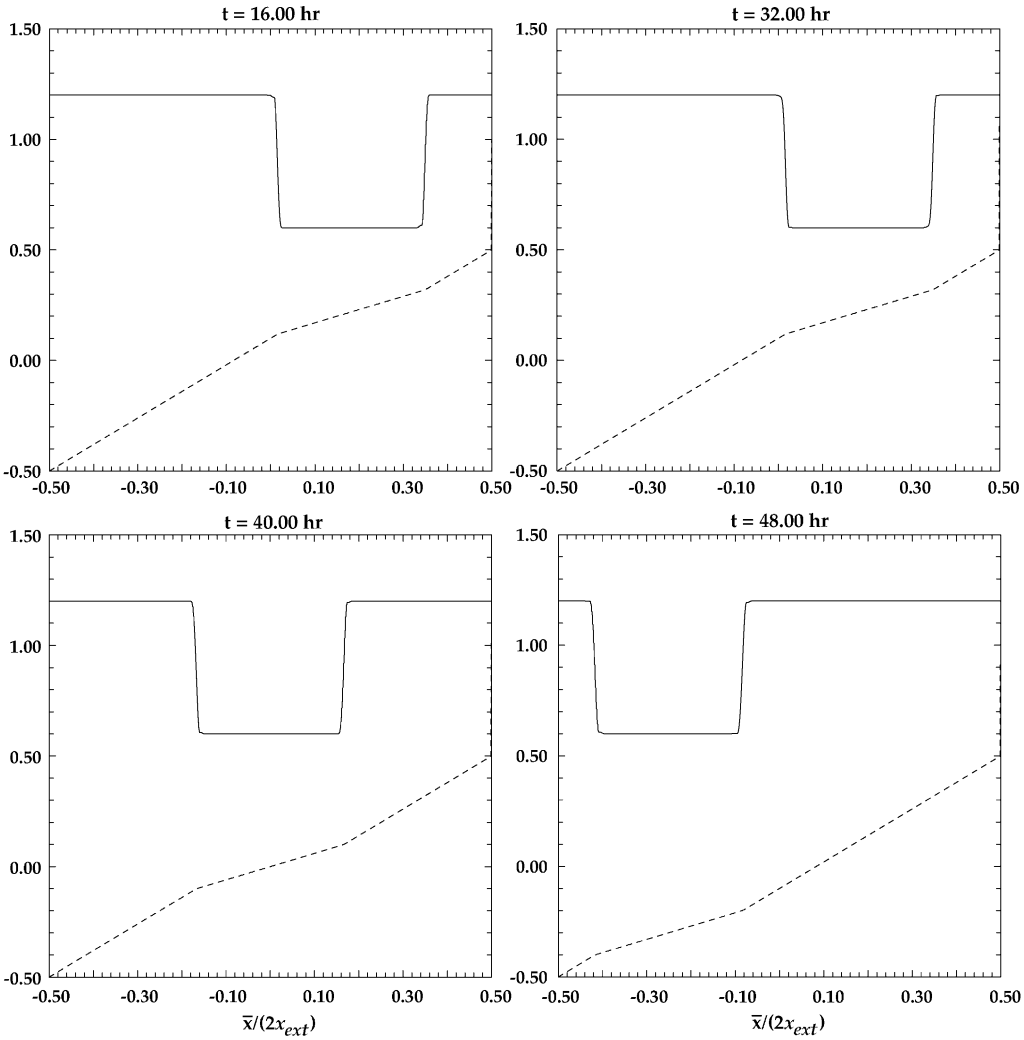

Fig. 3. Normalized grid stretching factor $\delta_x \Delta \bar{x}^{-1}$ (solid line), and normalized physical coordinate $x(\bar{t}, \bar{x})/(2x_{\text{ext}})$ (dashed line) as a function of normalized transformed coordinate for CDGA analogue of a "nested grid". Here $2x_{\text{ext}} = 5000$ km denotes the extent of the computational domain. See www.mmm.ucar.edu/asr2002/deepconv2.html for an animation of the results shown above and in the previous figure.

required to keep the computations stable. Termed VLES (for very-large-eddy simulation), our experience [44] is that this approach yields a physical accuracy at least as good as more traditional LES (large-eddy simulation) methods.

Fig. 4 illustrates the overall complexity of the flow. It shows instantaneous horizontal and vertical cross-sections, respectively, at the surface and equatorial plane, of the potential temperature (isentropes) θ, and zonal wind field u, after 3 years of simulated flow from case U0. Plates a and b depict the signature of baroclinic eddies at the surface (viz. weather systems), whereas plates a′ and b′ show the associated vertical structure with distinct stratification in the troposphere and stratosphere. The four plates of Fig. 5 depict the zonally averaged, 2.3-year means of the zonal wind field – a climatic feature – and contrast the complexity

Table 1
Summary of the four climate simulations

Simulation	Gridpoints	Type	Symmetry (%)	$V(\theta)_{\max}$	CPU time/δt
U0	64×32	Uniform	−8.3	$42.3K^2$	1
U2	64×64	Uniform	1.5	$49.6K^2$	1.6
SS	130×32	Stretched	2.0	$48.5K^2$	1.6
TS	64×32	Adaptive	−1.3	$46.7K^2$	1 Before, 0.8 after

The number of gridpoints refers to zonal × meridional points; all simulations used 41 vertical nodes with a vertical stepsize of 800 m, and a uniform time step of 900 s. Symmetry refers to the error in symmetry of the meridional wind field based upon the ratio of maximum to minimum zonally averaged, 2.3-year mean values. $V(\theta)_{\max}$ is the maximum value of the variation of the potential temperature θ, about its zonal-time average. The CPU time per time step is relative – the "before" and "after" values cited for the TS simulation refer to before and after the time adaptation. During the time adaptation the change was monotone.

of the instantaneous flow in Fig. 4. While a detailed comparison is outside the scope of this paper (see [44] for a more extended discussion), it is clear that each of the alternative simulations U0, SS, and TS offer some but not all of the features of the U2 simulation. Simulation SS provides the best match to U2 for the zonal flow in mid-troposphere ($z \sim 7$ km) in the equatorial band bounded by $\sim\pm 30°$, while TS matches better at somewhat higher latitudes but more poorly in the most central $\sim\pm 15°$ band. At high latitudes the U0 grid does better as the SS and TS grids have lower resolution near the poles. A similar pattern unfolds when comparing the instantaneous flows (not shown). Only the SS simulation shows the closed, westward jet high over the equator. The lower altitude, mid-latitude eastward jets appear too confined in latitude, however. Simulation TS shows these jets to be broader, though not to the same extent as shown in the U2 result.

Simple estimates of the accuracy of the simulations may be found by considering the symmetries of the flow. Assuming that the climate is stationary, then the zonally averaged 2.3-year means of the meridional wind should be anti-symmetric about the equator. Thus the magnitudes of the maximum and minimum values of meridional wind should be equal. The "symmetry" parameter in column 4 of Table 1 gives the departure of this ratio from unity. Simulations U2, SS, and TS all yield similar values with a departure from symmetry of $\sim$2%. The U0 symmetry error is four times larger. Another simple statistic, $V(\theta)_{\max}$ is given in column 5 of Table 1, it is the maximum value of the variation of potential temperature θ about its zonal-time average. For this statistic, the stretched and time adaptive simulations outperform the coarse, uniform grid simulation. This result generally holds for other global statistics based upon the variation of the flow, whereas statistics based solely on the zonal-time average do not show consistent improvement using SS and TS (with the exception of the vertical wind field).

A noteworthy result given in Table 1 is that use of CDGA *does not* necessarily increase CPU time. Consider the computational costs of the simulations as indicated in the last column of Table 1. This data shows that the stationary and time adapted grids (SS and TS, respectively) require 20% less CPU time per time step than the uniformly spaced grid (with the same number of grid points, U0). This result has been confirmed using an earlier, uniform grid version of the code. The decrease is due to accelerated convergence in the pressure solver – which we attribute to a decrease in the condition number of the elliptic operator. The reason for the improvement in condition number is that with larger meridional increments near the poles, the physical zonal increment Δx near the poles increases. The net effect is that by doubling the number of zonal nodes and by using stretching to enhance the meridional resolution in the equatorial region by a factor of two, we can simulate a fully doubled equatorial band resolution using only half as many nodes as needed for a uniform, higher resolution grid. Simultaneously, the computational time will be increased by only a factor of 1.6 compared to simulation U0. Typically, in a Cartesian domain, such an increase in resolution using a uniform grid (with four times as many nodes as in U0) would cost an order of magnitude more CPU time.

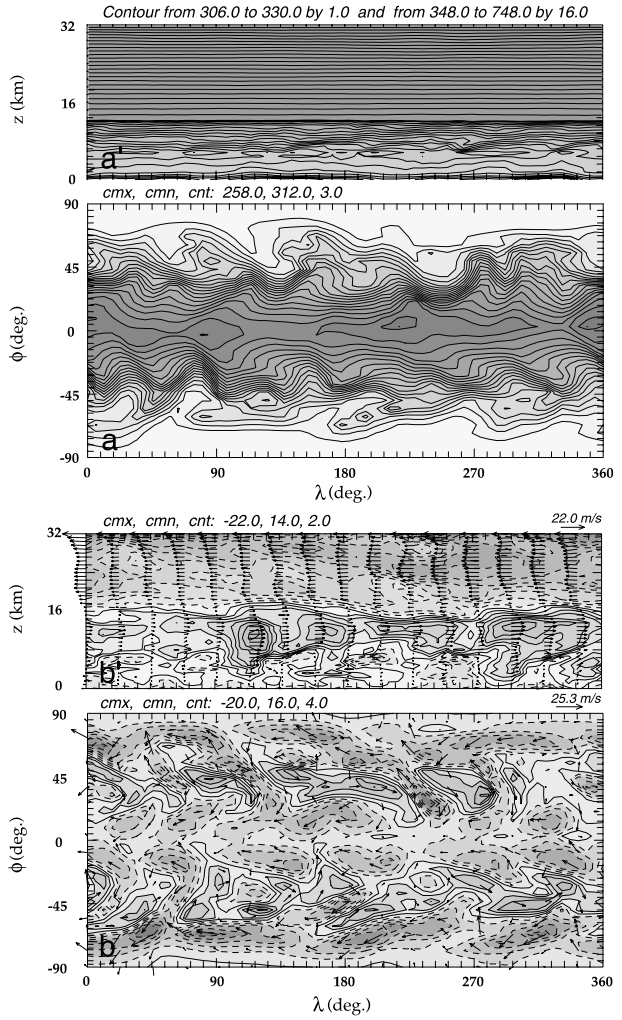

Fig. 4. Instantaneous solutions of the idealized climate problem after 3 years of simulation. Plates a′ and a show potential temperature field in the vertical equatorial plane and at the surface, respectively. Plates b′ and b display the zonal velocity contours with imposed flow vectors, respectively, in the equatorial plane and at the surface. Contour extrema and intervals are shown in the upper left corner of each plate (in plate a′ we used two contour increments to capture θ variability in the troposphere). Negative values are dashed. Maximum vector lengths are shown in the upper right corner of plates b′ and b. λ and ϕ denote the zonal (x/R_0) and meridional (y/R_0) angular coordinates, respectively, where R_0 is the radius of the Earth.

4. Remarks

The two example applications presented in this study demonstrate that a computational model that is designed from the bottom up combining NFT algorithms and generalized coordinates is ideally suited for the purpose of CDGA in a variety of geometrical systems. The use of generalized coordinates permits the model to work with spherical or rectangular computational domains with equal facility. Furthermore, it even could be used to model rapidly rotating, strongly oblate spheroidal bodies by using the terrain

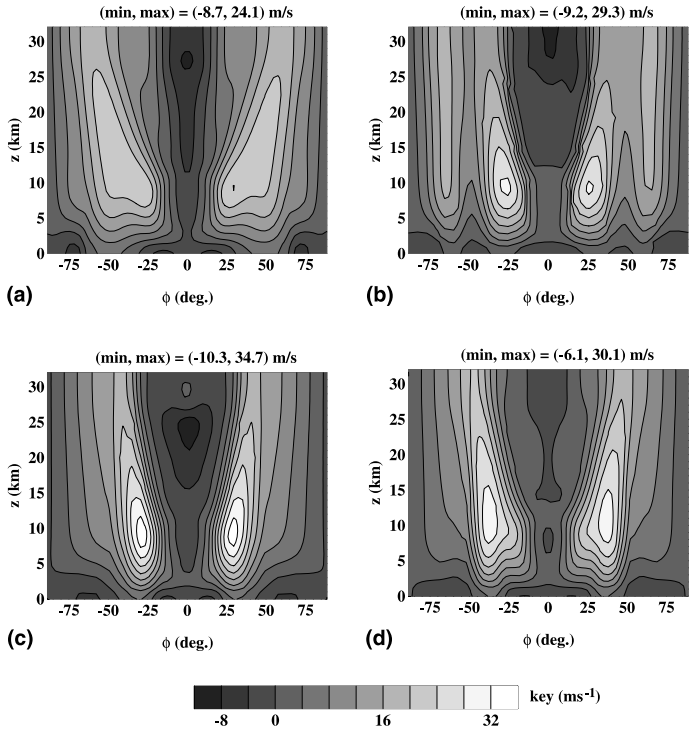

Fig. 5. Results from global simulation using coordinate transformations. Shown are zonally averaged, 2.3-year means of zonal winds. For convenience, min and max denote the extrema of the zonal wind fields; the same contours with an interval of 4.0 m s^{-1} are used in all plates. Contour magnitudes are given by the grayscale key. (a) Uniform horizontal grid U2; (b) uniform grid U0; (c) stationary stretched grid SS; (d) bi-modal time adaptive grid TS. Here ϕ denotes y/R_0.

following transformation to prescribe the shape of the spheroid. [11] The robust performance of NFT methods allow the generation of adaptive grids that are free of pathological oscillations (cf. [11] for a discussion). In particular, it enables the ability to mimic ''nested'' grids (NFT methods were designed with propagating discontinuities in mind). Finally, the ability of NFT methods to supply an effective implicit subgrid-scale model [44] facilitates LES studies in generalized coordinates by obviating the task of incorporating viscous stress.

Our model can handle with ease transformations presented either analytically or computed numerically. The computational overhead associated with analytical transformations are negligible and it is fairly straightforward to develop them when the regions of interest are few. For instance, applications such as a global study of equatorially trapped waves (e.g., Madden–Julian Oscillation [49]) may profit from the use of the class of analytic transformations, illustrated in Fig. 6:

$$Y(\overline{t},\overline{Y}) = S_y^{-1}\overline{Y} + \left(1 - S_y^{-1}\right)\overline{Y}^n, \tag{14}$$

$$X(\overline{t},\overline{X},\overline{Y}) = F_0(\overline{Y})\cdot X_0(\overline{t},\overline{X}) + F_1(\overline{Y})\cdot\overline{X} + T(\overline{t}), \tag{15}$$

[11] In geophysical/astrophysical applications adjustments are required in the environmental profiles to accommodate the resulting changes in the Coriolis and centrifugal force fields.

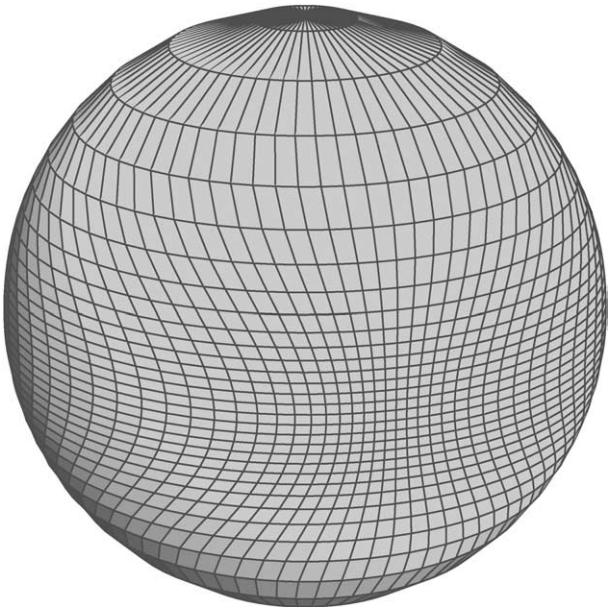

Fig. 6. Proposed time adaptive grid for study of equatorially trapped waves. This transformation is a generalization of SS from Section 3.2. Both magnitudes and locations of maximum stretching may change in response to variations of dependent fields. The "snapshot" shown here corresponds to a time when $S_x \equiv 3$ and $S_y \equiv 3$ in (14)–(17), with $n = 5$.

where, $\overline{X}$ and $\overline{Y}$ are normalized zonal and meridional coordinates such that $\overline{X} \in [0,1]$ and $\overline{Y} \in [-1,1]$. The auxiliary functions are:

$$F_0(\overline{Y}) = (1 - \overline{Y}^2), \quad F_1(\overline{Y}) = 1 - F_0(\overline{Y}), \tag{16}$$

$$X_0(\overline{t},\overline{X}) = \left[S_x^{-1} + \left(1 - S_x^{-1}\right) \cdot f(\overline{X})\right] \cdot \overline{X}, \tag{17}$$

where $f(s) = (10 - 15s + 6s^2) \cdot s^2$ and S_x^{-1}, S_y^{-1} are of the form $S^{-1}(\tau) = S_i^{-1} + (S_f^{-1} - S_i^{-1}) \cdot \tau \cdot f(\tau)$, for $\tau \in [0,1]$ where $\tau = (\overline{t} - \overline{t}_i)/(\overline{t} - \overline{t}_f)$, and S_i^{-1}, S_f^{-1} refer to the initial and final values of maximum stretch occurring at times $\overline{t}_i$, $\overline{t}_f$, respectively. For values of $\tau \notin [0,1]$, S^{-1} takes on constant value. For $\tau \in [0,1]$, S^{-1} is C^2 continuous. The function $T(\overline{t})$ in (15) allows zonal translation. SS results from (14)–(17) by choosing $S_x \equiv 1, S_y \equiv 2$, $n = 5$, and $T \equiv 0$.

Comparison of the four climate simulations in Section 3.2 suggests that substantially more sophisticated coordinate transformations (than above) may be required. For instance, it may be desirable for the time adapted stretching to trace the baroclinic eddies (evidenced in Fig. 4(a)). The general design of such analytic transformations that adapt to multiple physical features can be problematical. In the computational domain $\mathbf{S}_t$ (i.e., in the model code), it is the position of the physical coordinates as a function of grid indices (i.e., transformed coordinates) that defines all necessary metric coefficients and transformations. But the only a priori knowledge is that of the physical positions of the computational cells – viz. the knowledge of the transformed coordinates as a function of the physical coordinates. Thus, the problem (of designing suitable analytic transformations) is inherently implicit. In contrast, numerical evaluation of transformations that focus on tracing flow features can be technically easier – see the inertio-gravity-wave example in Section 3.1. Since the entire problem is posed in the transformed domain, there is no difficulty regards the implicitness of specified geometrical information. The transformation evolves in response to whatever flow features have been targeted.

Our focus to date has been to develop the model and test it using a few idealized cases/coordinate transformations. We have only begun to scratch the surface in terms of the type of transformations to use, especially in regards to numerical generation. We believe that the NFT advection of the grid increment – utilized in the wave propagation example – provides an important clue towards the direction such methods could go. Compared to more standard numerical transformations based upon the solutions of elliptic equations – e.g. [5,46] – this advection-based approach offers the advantages of: (i) computational efficiency, (ii) smoothness in time in addition to that in space, and (iii) a simplicity of imposing limits on minimum grid spacing. This last feature may be realized by advecting $\delta^* = \delta_x - \delta_{\min}$, rather than δ_x, where $\delta_{\min}$ is the minimum grid spacing allowed (corresponding to the maximum enhancement in resolution in $\mathbf{S}_p$). The positive definiteness of the NFT algorithm then assures that $\delta^* \geqslant 0$ for an arbitrary velocity of the mesh movement in (13). This is an important feature as it: (i) prevents the grid from collapsing into a singularity, (ii) minimizes Courant number limitations [19], and (iii) provides control on resolution enhancement that is independent of the number of grid points.

Acknowledgements

Comments from Robert Sharman and Joanna Szmelter on an earlier version of this paper, as well as discussions with Mikhail Shashkov, helped to improve the presentation. This work was supported in part by the Department of Energy "Climate Change Prediction Program" (CCPP) research initiative.

Appendix A. Elliptic pressure equation

The formulation of the elliptic pressure equation follows a standard projection procedure (cf. [6,7,20,39]). Here, the discrete counterparts (12) of Eqs. (6) and (7) are first manipulated algebraically to provide closed form formulae for the physical velocities v^j. This is straightforward due to the nonstaggered grid employed. The resulting formulae are then inserted into the discrete version of (5) utilizing (9). This leads to the elliptic pressure equation

$$-\frac{\Delta t}{\rho^*}\frac{\partial}{\partial \overline{x}^j}\left[\rho^* \mathscr{E}\left(\widetilde{\mathscr{V}}^j - \widetilde{\mathscr{C}}^{jk}\frac{\partial \pi'}{\partial \overline{x}^k}\right)\right] = 0, \tag{A.1}$$

where the entire equation has been premultiplied by $-\Delta t/\rho^*$. The factor (-1) assures the formal negative-definiteness of the elliptic operator on the lhs of (A.1); further normalization by $\Delta t/(\rho^*)$ gives the residual errors of (A.1) the sense of the divergence of a dimensionless velocity on the grid. [12] The latter compares directly to the magnitudes of the Courant and Lipschitz numbers and facilitates the design of a physically meaningful stopping criteria [38]. The functions $\widetilde{\mathscr{V}}^j$ and $\widetilde{\mathscr{C}}^{jk}$ are

$$\widetilde{\mathscr{V}}^j = \widetilde{G}^j_p \mathscr{V}^p, \tag{A.2}$$

$$\widetilde{\mathscr{C}}^{jk} = \widetilde{G}^j_p \mathscr{C}^{pk}, \tag{A.3}$$

where $p = 1, 2, 3$; $\mathscr{V}^p$ and $\mathscr{C}^{pk}$ are defined below.

[12] Note that $\mathscr{E}(\widetilde{\mathscr{V}}^j - \cdots)$ in (A.1) is the updated solenoidal velocity $\overline{v^j}|^{n+1}$, so the impermeability condition $\overline{v^3}|^{n+1} = 0$ at the model surface and lid, translates via (8) into the implicit Neumann boundary conditions for pressure $\widetilde{\mathscr{V}}^3 = \widetilde{\mathscr{C}}^{3k}(\partial \pi'/\partial x^k) - \mathscr{E}^{-1}(\partial \overline{x}^3/\partial t)$; see [37,42] for discussion.

If $\mathbf{S}_p$ is chosen to be a spherical coordinate system centered on a rotating planet, the nonsymmetric coefficients $\widetilde{G}_p^j$ take on the values: $\widetilde{G}_1^1 = E_{,x}/(\Gamma\cos\phi)$, $\widetilde{G}_1^2 = D_{,x}/(\Gamma\cos\phi)$, $\widetilde{G}_1^3 = C_{,x}/(\Gamma\cos\phi)$, $\widetilde{G}_2^1 = E_{,y}/\Gamma$, $\widetilde{G}_2^2 = D_{,y}/\Gamma$, $\widetilde{G}_2^3 = C_{,y}/\Gamma$, $\widetilde{G}_3^1 \equiv 0$, $\widetilde{G}_3^2 \equiv 0$, and $\widetilde{G}_3^3 = C_{,z}$. Subscripts preceded by a comma denote partial differentiation. $\Gamma = r/R_0$, where R_0 is the planetary radius, and (r,ϕ) denote radius and latitude, respectively. In the limit $R_0 \to \infty$, $\Gamma \to 1$. The geometry of $\mathbf{S}_p$ then becomes a Cartesian "Beta-" or "F-plane". With this choice of $\mathbf{S}_p$ (i.e, dual spherical/Cartesian system), the coefficient $\mathscr{E}$ becomes

$$\mathscr{E} = \left[-\mathscr{G}\mathscr{F}_2\vartheta_x + \mathscr{G}\mathscr{F}_2\mathscr{F}_3\vartheta_y + \mathscr{G}\left(1+\mathscr{F}_3^2\right)\vartheta_z + (1+\widetilde{\alpha}^*)\left(1+\mathscr{F}_2^2+\mathscr{F}_3^2\right)\right]^{-1}, \tag{A.4}$$

where $\mathscr{G} = 0.5\Delta t(g/\theta_b)(1+\alpha^*)^{-1}$ and $\vartheta_j = 0.5\Delta t\widetilde{G}_j^k\theta_{e,x^k}$. Here $\widetilde{\alpha}^* = 0.5\Delta t\widetilde{\alpha}$ and $\alpha^* = 0.5\Delta t\alpha$. The parameters α and $\widetilde{\alpha}$ are the inverse damping times for absorbers in the vicinity of boundaries for the momentum (6) and potential temperature Eq. (7), respectively. The parameters $\mathscr{F}_j = 0.5\Delta t f_j(1+\alpha^*)^{-1}$, where f_j are the components of the geophysical Coriolis force (placed in the F^j term of Eq. (6)). In the dual spherical/Cartesian system, $f_1 \equiv 0$, $f_2 = 2\Omega\cos\phi$, $f_3 = 2\Omega\sin\phi$, and Ω is the planetary rotation rate. [13] Such a dual $\mathbf{S}_p$ is now assumed for the remaining appendix variables.

The auxiliary fields $\mathscr{V}^j$ used in Eq. (A.2) can be compactly written as

$$\mathscr{V}^1 = \mathscr{A}U + \mathscr{B}V - \mathscr{R}\mathscr{F}_2W, \tag{A.5}$$

$$\mathscr{V}^2 = \mathscr{C}U + \mathscr{D}V + \mathscr{R}\mathscr{F}_2\mathscr{F}_3W, \tag{A.6}$$

$$\mathscr{V}^3 = \mathscr{H}U + \mathscr{I}V + \mathscr{R}\left(1+\mathscr{F}_3^2\right)W, \tag{A.7}$$

where the coefficients $\mathscr{A}$ to $\mathscr{I}$ are equal to

$$\mathscr{A} = \mathscr{R} + (1+\alpha^*)^{-1}\mathscr{G}\vartheta_z, \tag{A.8}$$

$$\mathscr{B} = \mathscr{R}\mathscr{F}_3 + (1+\alpha^*)^{-1}\mathscr{G}(\mathscr{F}_2\vartheta_y + \mathscr{F}_3\vartheta_z), \tag{A.9}$$

$$\mathscr{C} = -\mathscr{R}\mathscr{F}_3 - (1+\alpha^*)^{-1}\mathscr{G}\mathscr{F}_3\vartheta_z, \tag{A.10}$$

$$\mathscr{D} = \mathscr{R}\left(1+\mathscr{F}_2^2\right) + (1+\alpha^*)^{-1}\mathscr{G}(-\mathscr{F}_2\vartheta_x + \vartheta_z), \tag{A.11}$$

$$\mathscr{H} = \mathscr{R}\mathscr{F}_2 + (1+\alpha^*)^{-1}\mathscr{G}(-\vartheta_x + \mathscr{F}_3\vartheta_y), \tag{A.12}$$

$$\mathscr{I} = \mathscr{R}\mathscr{F}_2\mathscr{F}_3 - (1+\alpha^*)^{-1}\mathscr{G}(\mathscr{F}_3\vartheta_x + \vartheta_y), \tag{A.13}$$

and the velocities U, V, and W are defined as

$$U \equiv \mathrm{LE}(\widetilde{u}) + 0.5\Delta t(\alpha u_e - f_3 v_e + f_2 w_e), \tag{A.14}$$

$$V \equiv \mathrm{LE}(\widetilde{v}) + 0.5\Delta t(\alpha v_e + f_3 u_e), \tag{A.15}$$

[13] Strictly speaking, these values of f_j require that $\mathbf{S}_p$ be *aligned* with the rotation axis; for a spherical system, this axis must pass through the coordinate poles.

$$W \equiv \mathrm{LE}(\widetilde{w}) + \tilde{\mathscr{G}}\tilde{\theta}' + 0.5\Delta t(\alpha w_e - f_2 u_e), \tag{A.16}$$

where $\tilde{\theta}' := \tilde{\theta} - \theta_e$ (see Eq. (12) for the definition of $\tilde{\theta}$), $\tilde{\mathscr{G}} := 0.5\Delta t g \theta_b^{-1}(1+\widetilde{\alpha}^*)^{-1}$, and $\mathscr{R} := (1+\widetilde{\alpha}^*)(1+\alpha^*)^{-1}$.

The coefficients $\mathscr{C}^{jk}$ used in Eq. (A.3) take the explicit form:

$$\mathscr{C}^{11} = \mathscr{R}\Big(\widetilde{G}_1^1 + \widetilde{G}_2^1\mathscr{F}_3\Big) + (1+\alpha^*)^{-1}\mathscr{G}\Big[\widetilde{G}_2^1\mathscr{F}_2\vartheta_y + \Big(\widetilde{G}_1^1 + \widetilde{G}_2^1\mathscr{F}_3\Big)\vartheta_z\Big], \tag{A.17}$$

$$\mathscr{C}^{12} = \mathscr{R}\Big(\widetilde{G}_1^2 + \widetilde{G}_2^2\mathscr{F}_3\Big) + (1+\alpha^*)^{-1}\mathscr{G}\Big[\widetilde{G}_2^2\mathscr{F}_2\vartheta_y. + \Big(\widetilde{G}_1^2 + \widetilde{G}_2^2\mathscr{F}_3\Big)\vartheta_z\Big], \tag{A.18}$$

$$\mathscr{C}^{13} = \mathscr{R}\Big(\widetilde{G}_1^3 + \widetilde{G}_2^3\mathscr{F}_3 - \widetilde{G}_3^3\mathscr{F}_2\Big) + (1+\alpha^*)^{-1}\mathscr{G}\Big[\widetilde{G}_2^3\mathscr{F}_2\vartheta_y + \Big(\widetilde{G}_1^3 + \widetilde{G}_2^3\mathscr{F}_3\Big)\vartheta_z\Big], \tag{A.19}$$

$$\mathscr{C}^{21} = \mathscr{R}\Big[-\widetilde{G}_1^1\mathscr{F}_3 + \widetilde{G}_2^1\Big(1+\mathscr{F}_2^2\Big)\Big] + (1+\alpha^*)^{-1}\mathscr{G}\Big[-\widetilde{G}_2^1\mathscr{F}_2\vartheta_x + \Big(-\widetilde{G}_1^1\mathscr{F}_3 + \widetilde{G}_2^1\Big)\vartheta_z\Big], \tag{A.20}$$

$$\mathscr{C}^{22} = \mathscr{R}\Big[-\widetilde{G}_1^2\mathscr{F}_3 + \widetilde{G}_2^2\Big(1+\mathscr{F}_2^2\Big)\Big] + (1+\alpha^*)^{-1}\mathscr{G}\Big[-\widetilde{G}_2^2\mathscr{F}_2\vartheta_x + \Big(-\widetilde{G}_1^2\mathscr{F}_3 + \widetilde{G}_2^2\Big)\vartheta_z\Big], \tag{A.21}$$

$$\mathscr{C}^{23} = \mathscr{R}\Big[-\widetilde{G}_1^3\mathscr{F}_3 + \widetilde{G}_2^3(1+\mathscr{F}_2^2) + \widetilde{G}_3^3\mathscr{F}_2\mathscr{F}_3\Big] + (1+\alpha^*)^{-1}\mathscr{G}\Big[-\widetilde{G}_2^3\mathscr{F}_2\vartheta_x + \Big(-\widetilde{G}_1^3\mathscr{F}_3. + \widetilde{G}_2^3\Big)\vartheta_z\Big], \tag{A.22}$$

$$\mathscr{C}^{31} = \mathscr{R}\mathscr{F}_2\Big(\widetilde{G}_1^1 + \widetilde{G}_2^1\mathscr{F}_3\Big) + (1+\alpha^*)^{-1}\mathscr{G}\Big[-\Big(\widetilde{G}_1^1 - \widetilde{G}_2^1\mathscr{F}_3\Big)\vartheta_x + \Big(\widetilde{G}_1^1\mathscr{F}_3 - \widetilde{G}_2^1\Big)\vartheta_y\Big], \tag{A.23}$$

$$\mathscr{C}^{32} = \mathscr{R}\mathscr{F}_2\Big(\widetilde{G}_1^2 + \widetilde{G}_2^2\mathscr{F}_3\Big) + (1+\alpha^*)^{-1}\mathscr{G}\Big[-\Big(\widetilde{G}_1^2 + \widetilde{G}_2^2\mathscr{F}_3\Big)\vartheta_x + \Big(\widetilde{G}_1^2\mathscr{F}_3 - \widetilde{G}_2^2\Big)\vartheta_y\Big], \tag{A.24}$$

$$\mathscr{C}^{33} = \mathscr{R}\Big[\widetilde{G}_1^3\mathscr{F}_2. + \widetilde{G}_2^3\mathscr{F}_2\mathscr{F}_3 + \widetilde{G}_3^3(1+\mathscr{F}_3^2)\Big] + (1+\alpha^*)^{-1}\mathscr{G}\Big[-\Big(\widetilde{G}_1^3 + \widetilde{G}_2^3\mathscr{F}_3\Big)\vartheta_x + \Big(\widetilde{G}_1^3\mathscr{F}_3 - \widetilde{G}_2^3\Big)\vartheta_y\Big]. \tag{A.25}$$

Appendix B. Analytical nested grid transformation

Here we develop the analytical representation of the mapping function for the nested grid simulation in Section 3.1. We begin by defining $\delta_x' := \partial x/\partial\bar{x}$ that is, in general, the analytic limit of $\delta_x\Delta\bar{x}^{-1}$. By design of the experiment, the physical-grid-increment function $\delta_x(\bar{t},\bar{x})$ is a translating step function so one anticipates

$$\delta_x'(\bar{t},\bar{x}) = \gamma\{p - (p-1)[H(\bar{x} - \bar{x}_L(\bar{t})) - H(\bar{x} - \bar{x}_R(\bar{t}))]\}, \tag{B.1}$$

where $H(s)$ is the Heaviside distribution, parameter p is the ratio of the outer (coarse) and inner (high resolution) δ_x', and γ is the normalization factor assuring that the physical and transformed domains are of equal size. The functions $\bar{x}_L$ and $\bar{x}_R$ represent the left and right endpoints of the transformed interval of enhanced resolution, respectively. In the physical domain, we assume that this interval corresponds to $[x_L = x_d - l, x_d + l = x_R]$ where $x_d = x_d(t)$ gives the location of the center of the forcing disturbance and l is

the half-width of the interval. The speed at which the interval translates (in physical coordinates) is uniform and is given by $\mathrm{d}x_d/\mathrm{d}t := \dot{x}_d$. Given the forcing used in Section 3.1, $z_s(t, x_d(t))$ always gives the minimum value of z_s, and $\dot{x}_d = 2\pi\mathscr{L}/T\cos(2\pi t/T)$.

In order to determine the value of γ, as well as the length of the high-resolution interval in *transformed* coordinates, denoted as $2\bar{l}$, we integrate the definition of δ'_x

$$x(\bar{x}) = -x_{\text{ext}} + \int_{-x_{\text{ext}}}^{\bar{x}} \delta'_s \,\mathrm{d}\bar{s}. \tag{B.2}$$

Here $\pm x_{\text{ext}}$ denotes the endpoints of the computational domain in *either* physical or transformed coordinates, and s is a dummy integration variable. Setting: (i) $\bar{x} = \bar{x}_L$, (ii) and then $\bar{x} = \bar{x}_R$, (iii) substituting each into (B.2), and (iv) subtracting the two piecewise integrated equations yields the result $\bar{l} = l/\gamma$. Next, we set $\bar{x} = x_{\text{ext}}$ and integrate (B.2) to find $\gamma = x_{\text{ext}}/(2x_{\text{ext}} - \bar{l})$. These last two relationships are solved to find

$$\gamma = \frac{(p-1)l + x_{\text{ext}}}{p x_{\text{ext}}}, \tag{B.3}$$

$$\bar{l} = \frac{p l x_{\text{ext}}}{(p-1)l + x_{\text{ext}}}. \tag{B.4}$$

By integrating (B.2) to any point in $[\bar{x}_L, \bar{x}_R]$ and then taking the partial derivative with respect to $\bar{t}$, it follows that the speed of propagation of the forcing disturbance and inner grid in transformed coordinates is $\partial\bar{x}_d/\partial\bar{t} = \dot{x}_d/(p\gamma)$. This completes the construction of an analytical solution for δ'_x.

Finally we demonstrate that (B.1) is also the exact solution to the mesh-continuity Eq. (13) by direct substitution and differentiation. Assuming $U = U(\bar{t})$ (i.e., U is uniform in $\bar{x}$), and noting that $\partial H(s)/\partial s = \delta_D(s)$ where δ_D is the Dirac delta distribution, we find the divergence term of (13) is:

$$\frac{\partial U\delta'_x}{\partial\bar{x}} = U\frac{\partial\delta'_x}{\partial\bar{x}} = -U(p-1)\gamma[\delta_D(\bar{x}-\bar{x}_L) - \delta_D(\bar{x}-\bar{x}_R)]. \tag{B.5}$$

The transient term in (13) is evaluated in similar fashion noting that $\bar{x}_L, \bar{x}_R$ are "timelike", e.g., $\partial H(\bar{x}-\bar{x}_L)/\partial\bar{t} = -\delta_D(\bar{x}-\bar{x}_L)\partial\bar{x}_L/\partial\bar{t}$. Thus:

$$\frac{\partial\delta'_x}{\partial\bar{t}} = ((p-1)/p)\dot{x}_d[\delta_D(\bar{x}-\bar{x}_L) - \delta_D(\bar{x}-\bar{x}_R)]. \tag{B.6}$$

Comparison of (13), (B.5), and (B.6) immediately leads to the result that $U(\bar{t}(t)) = \dot{x}_d/(p\gamma) = \beta\dot{x}_d$ where

$$\beta = x_{\text{ext}}/[x_{\text{ext}} + (p-1)l]. \tag{B.7}$$

For the doubly enhanced resolution case discussed in Section 3.1; $p = 2$, $l = 500$ km, and $x_{\text{ext}} = 2500$ km. This yields $\gamma = 3/5$, $\bar{l} = 833.33\ldots$ km (normalized value $= 1/6$), and $\beta = 5/6$. The location of the moving nested grid may be determined by integrating (B.2), e.g., at 48 h, $\bar{x}_L = -x_{\text{ext}} + (x_L + x_{\text{ext}})/(2\gamma) = -2083.33\ldots$ km (normalized value $= -5/12$). These values are in perfect agreement with Fig. 3.

References

[1] D.A. Anderson, Adaptive grid methods for partial differential equations, in: K.N. Ghia, U. Ghia (Eds.), Advances in Grid Generation, Fed-5, American Society of Mechanical Engineers, 1983, pp. 1–15.
[2] Policy Statement, Bull. Am. Meteor. Soc. 81 (2000) 1341.

[3] L. Boltzmann, Zur integration der diffusiongleichung bei variabeln diffusionscoefficienten, Ann. Phys. Chem. S. 3 (1894) 959.
[4] D.P. Bacon, N.N. Ahmad, Z. Boybeyi, T.J. Dunn, M.S. Hall, P.C.S. Lee, R.A. Sarma, M.D. Turner, A dynamically adapting weather and dispersion model: the operational multiscale environment model with grid adaptivity (OMEGA), Mon. Weather Rev. 128 (2000) 2044–2076.
[5] J.U. Brackbill, J.S. Saltzman, Adaptive rezoning and singular problems in two dimensions, J. Comput. Phys. 46 (1982) 342–368.
[6] A.J. Chorin, Numerical solution of the Navier–Stokes equations, Math. Comput. 22 (1968) 742–762.
[7] T.L. Clark, A small-scale dynamic model using a terrain-following coordinate transformation, J. Comput. Phys. 24 (1977) 186–214.
[8] T.L. Clark, R.D. Farley, Severe downslope windstorm calculations in two and three spatial dimensions using anelastic interactive grid nesting: a possible mechanism for gustiness, J. Atmos. Sci. 41 (1984) 329–350.
[9] C.S. Cotter, P.K. Smolarkiewicz, I.N. Szczyrba, A viscoelastic model for brain injuries, Int. J. Numer. Meth. Fluids 40 (2002) 303–311.
[10] M. Cullen, D. Salmond, P.K. Smolarkiewicz, Key numerical issues for future development of the ECMWF model, in: Proc. ECMWF Workshop on Developments in Numerical Methods for Very High Resolution Global Models, 5–7 June 2000, Reading, UK, ECMWF, 2000, pp. 183–206.
[11] G.S. Dietachmayer, K.K. Droegemeier, Application of continuous dynamic grid adaptation techniques to meteorological modeling. Part I: Basic formulation and accuracy, Mon. Weather Rev. 120 (1992) 1675–1706.
[12] J.R. Elliott, P.K. Smolarkiewicz, Eddy resolving simulations of turbulent solar convection, Int. J. Numer. Meth. Fluids 39 (2002) 855–864.
[13] B.H. Fiedler, R.J. Trapp, A fast dynamic grid adaptation scheme for meteorological flows, Mon. Weather Rev. 121 (1992) 2879–2888.
[14] M.S. Fox-Rabinovitz, G.L. Stenchikov, M.J. Suarez, L.L. Takacs, A finite-difference GCM dynamical core with a variable-resolution stretched grid, Mon. Weather Rev. 125 (1997) 2943–2968.
[15] M.S. Fox-Rabinovitz, G.L. Stenchikov, M.J. Suarez, L.L. Takacs, R.C. Govindaraju, A uniform and variable resolution stretched-grid GCM dynamical core with realistic orography, Mon. Weather Rev. 128 (2000) 1883–1898.
[16] T. Gal-Chen, C.J. Somerville, On the use of a coordinate transformation for the solution of the Navier–Stokes equations, J. Comput. Phys. 17 (1975) 209–228.
[17] W.W. Grabowski, P.K. Smolarkiewicz, A multiscale anelastic model for meteorological research, Mon. Weather Rev. 130 (2002) 939–956.
[18] I.M. Held, M.J. Suarez, A proposal for intercomparison of the dynamical cores of atmospheric general circulation models, Bull. Am. Meteorol. Soc. 75 (1994) 1825–1830.
[19] J.P. Iselin, J.M. Prusa, W.J. Gutowski, Dynamic grid adaptation using the MPDATA scheme, Mon. Weather Rev. 130 (2002) 1026–1039.
[20] H. Kapitza, D. Eppel, The non-hydrostatic mesoscale model GESIMA. Part 1: dynamical equations and tests, Beitr. Phys. Atmos. 65 (1992) 129–146.
[21] H.G. Landau, Heat conduction in a melting solid, Q. Appl. Math. 8 (1949) 81.
[22] F.B. Lipps, R.S. Hemler, A scale analysis of deep moist convection and some related numerical calculations, J. Atmos. Sci. 39 (1982) 2192–2210.
[23] L.G. Margolin, P.K. Smolarkiewicz, Antidiffusive velocities for multipass donor cell advection, SIAM J. Sci. Comput. 20 (3) (1998) 907–929.
[24] L.G. Margolin, P.K. Smolarkiewicz, Z. Sorbjan, Large-eddy simulations of convective boundary layers using nonoscillatory differencing, Physica D 133 (1999) 390–397.
[25] J. Marshall, A. Adcroft, C. Hill, L. Perelman, C. Heisey, A finite-volume incompressible Navier–Stokes model for studies of the ocean on parallel computers, J. Geophys. Res. 102 (C3) (1997) 5753–5766.
[26] C.W. Misner, K.S. Thorne, J.A. Wheeler, in: Gravitation, W.H. Freeman, San Francisco, 1973, p. 1279.
[27] S.M. Polavarapu, W.R. Peltier, The structure and nonlinear evolution of synoptic scale cyclones: life cycle simulations with a cloud-scale model, J. Atmos. Sci. 47 (1990) 2645–2673.
[28] L. Prandtl, Aurberechung der grenzxchichten, Z. Angew. Math. 18 (1938) 77, Also in Laminar Boundary Layers, by L. Rosenhead, Oxford University Press, London, 1963.
[29] J.M. Prusa, P.K. Smolarkiewicz, R.R. Garcia, On the propagation and breaking at high altitudes of gravity waves excited by tropospheric forcing, J. Atmos. Sci. 53 (1996) 2186–2216.
[30] J.M. Prusa, P.K. Smolarkiewicz, A.A. Wyszogrodzki, Simulations of gravity wave induced turbulence using 512 PE CRAY T3E, Int. J. Appl. Math. Comput. Sci. 11 (4) (2001) 101–115.
[31] R.D. Sharman, T.L. Keller, M.G. Wurtele, Incompressible and anelastic flow simulations on numerically generated grids, Mon. Weather Rev. 116 (1988) 1124–1136.
[32] W.C. Skamarock, J. Oliger, R.L. Street, Adaptive grid refinement for numerical weather prediction, J. Comput. Phys. 80 (1989) 27–60.

[33] W.C. Skamarock, P.K. Smolarkiewicz, J.B. Klemp, Preconditioned conjugate-residual solvers for Helmholtz equations in nonhydrostatic models, Mon. Weather Rev. 125 (1997) 587–599.
[34] P.K. Smolarkiewicz, G.A. Grell, A class of monotone interpolation schemes, J. Comput. Phys. 101 (1992) 431–440.
[35] P.K. Smolarkiewicz, J.A. Pudykiewicz, A class of semi-Lagrangian approximations for fluids, J. Atmos. Sci. 49 (1992) 2082–2096.
[36] P.K. Smolarkiewicz, L.G. Margolin, On forward-in-time differencing for fluids: extension to a curvilinear framework, Mon. Weather Rev. 121 (1993) 1847–1859.
[37] P.K. Smolarkiewicz, L.G. Margolin, Variational solver for elliptic problems in atmospheric flows, Appl. Math. Comput. Sci. 4 (1994) 527–551.
[38] P.K. Smolarkiewicz, V. Grubišić, L.G. Margolin, On forward-in-time differencing for fluids: stopping criteria for iterative solutions of anelastic pressure equations, Mon. Weather Rev. 25 (1997) 647–654.
[39] P.K. Smolarkiewicz, L.G. Margolin, On forward-in-time differencing for fluids: an Eulerian/semi-Lagrangian nonhydrostatic model for stratified flows, Atmos. Ocean Special 35 (1997) 127–152.
[40] P.K. Smolarkiewicz, L.G. Margolin, MPDATA: a finite-difference solver for geophysical flows, J. Comput. Phys. 140 (1998) 459–480.
[41] P.K. Smolarkiewicz, V. Grubišić, L.G. Margolin, A.A. Wyszogrodzki, Forward-in-time differencing for fluids: nonhydrostatic modeling of fluid motions on a sphere, in: Proc. 1998 Seminar on Recent Developments in Numerical Methods for Atmospheric Modeling, Reading, UK, ECMWF, 1999, pp. 21–43.
[42] P.K. Smolarkiewicz, L.G. Margolin, Variational methods for elliptic problems in fluid models, in: Proc. ECMWF Workshop on Developments in Numerical Methods for Very High Resolution Global Models, 5–7 June 2000, Reading, UK, ECMWF, 2000, pp. 137–159.
[43] P.K. Smolarkiewicz, L.G. Margolin, A.A. Wyszogrodzki, A class of nonhydrostatic global models, J. Atmos. Sci. 58 (2001) 349–364.
[44] P.K. Smolarkiewicz, J.M. Prusa, Forward-in-time differencing for fluids: simulation of geophysical turbulence, in: D. Drikakis, B.J. Guertz (Eds.), Turbulent Flow Computation, Kluwer Academic Publishers, Dordrecht, 2002, pp. 207–240.
[45] J.L. Synge, A. Schild, in: Tensor Calculus, Dover Press, New York, 1978, p. 324.
[46] J.F. Thompson, F.C. Thames, C.W. Mastin, Automatic numerical generation of body-fitted curvilinear coordinate system for field containing any number of arbitrary two-dimensional bodies, J. Comput. Phys. 15 (1974) 299–319.
[47] J. Wu, J.Z. Zhu, J. Szmelter, O.C. Zienkiewicz, Error estimation and adaptivity in Navier–Stokes incompressible flows, Comput. Mech. 6 (1990) 259–270.
[48] M. Vinokur, Conservation equations of gasdynamics in curvilinear coordinate systems, J. Comput. Phys. 14 (1974) 105–125.
[49] J.-I. Yano, J.C. McWilliams, M.W. Moncrieff, Hierarchical tropical cloud systems in an analog shallow-water model, J. Atmos. Sci. 52 (1995) 1723–1742.