

# DCMIP2016, Part 1: Models and Equation Sets

Paul A. Ullrich<sup>1</sup>, Christiane Jablonowski<sup>2</sup>, James Kent<sup>3</sup>, Peter H. Lauritzen<sup>4</sup>, Ramachandran Nair<sup>4</sup>, Kevin A. Reed<sup>5</sup>, Colin M. Zarzycki<sup>4</sup>, David A. Hall<sup>6</sup>, Alex Reinecke<sup>7</sup>, Kevin Viner<sup>7</sup>, Don Dazlich<sup>8</sup>, Ross Heikes<sup>8</sup>, Celal Konor<sup>8</sup>, David Randall<sup>8</sup>, Xi Chen<sup>9</sup>, Lucas Harris<sup>9</sup>, Marco Giorgetta<sup>10</sup>, Daniel Reinert<sup>11</sup>, Christian Kühnlein<sup>12</sup>, Robert Walko<sup>13</sup>, Vivian Lee<sup>14</sup>, Abdessamad Qaddouri<sup>14</sup>, Claude Girard<sup>14</sup>, Hiroaki Miura<sup>15</sup>, Tomoki Ohno<sup>15</sup>, Ryuji Yoshida<sup>16</sup>, Joseph Klemp<sup>4</sup>, Sang-Hun Park<sup>4</sup>, William Skamarock<sup>4</sup>, Thomas Dubos<sup>17</sup>, and Yann Meurdesoif<sup>17</sup>

<sup>1</sup>University of California, Davis

<sup>2</sup>University of Michigan

<sup>3</sup>University of South Wales

<sup>4</sup>National Center for Atmospheric Research

<sup>5</sup>Stony Brook University

<sup>6</sup>University of Colorado, Boulder

<sup>7</sup>Naval Research Laboratory

<sup>8</sup>Colorado State University

<sup>9</sup>Geophysical Fluid Dynamics Laboratory

<sup>10</sup>Max Planck Institute for Meteorology

<sup>11</sup>Deutscher Wetterdienst (DWD)

<sup>12</sup>European Centre for Medium-Range Weather Forecasts

<sup>13</sup>University of Miami

<sup>14</sup>Environment and Climate Change Canada

<sup>15</sup>University of Tokyo

<sup>16</sup>RIKEN

<sup>17</sup>Institut Pierre-Simon Laplace (IPSL)

<sup>18</sup>Lawrence Berkeley National Laboratory

*Correspondence to:* Paul A. Ullrich (paulullrich@ucdavis.edu)

**Abstract.** Atmospheric dynamical cores are a core component of global atmospheric modeling systems, and are responsible for capturing the dynamical behavior of the Earth's atmosphere via numerical integration of the Navier-Stokes equations. These systems have existed in one form or another for over half of a century, with the earliest strategies now evolved into a complex ecosystem of algorithms and computational strategies. In essence, no two dynamical cores are alike, and their individual successes suggest that no perfect model exists. To better understand modern dynamical cores, this paper aims to provide a comprehensive review of their design, including model grids, relevant equation sets, numerical stabilization techniques and idealized physics routines. Examples are taken from operational dynamical cores that participated in the 2016 Dynamical Core Model Intercomparison Project (DCMIP) workshop and summer school.

## 1 Introduction

The Dynamical Core Model Intercomparison Project (DCMIP) is an ongoing effort targeting the intercomparison of a fundamental component of global atmospheric modeling systems: the dynamical core. Although this component's role is simply to solve the equations of fluid motion (the Naviér-Stokes equations) governing atmospheric dynamics, there are numerous confounding factors that arise as a consequence of compromises that are required to make simulation computationally feasible. These factors include the choice of model grid, choice of prognostic equations, vertical coordinates, representation of topography, numerical method, temporal discretization, physics/dynamics coupling frequency, and the manner in which artificial diffusion, filters and/or energy/mass fixers are applied.

To advance the intercomparison project and provide a unique educational opportunity for students, DCMIP hosted a multidisciplinary two-week summer school and model intercomparison project, held at the National Center for Atmospheric Research (NCAR) in June 2016, that invited graduate students, postdocs, atmospheric modelers, expert lecturers and computer specialists to create a stimulating, unique and hands-on driven learning environment. It was built on earlier DCMIP workshops (held in 2008 and 2012), and previous intercomparison efforts by addressing key outstanding issues in global atmospheric models, incorporate international participation, and provide a unique training experience for the future generation of climate scientists. Special attention was paid to the role of simplified physical parameterizations, physics-dynamics coupling, non-hydrostatic atmospheric modeling and variable-resolution global modeling. The summer school and model intercomparison project promoted active learning, innovation, discovery, mentorship and the integration of science and education. Modeling groups were then invited to contribute model descriptions and results to the intercomparison effort for publication.

The summer school directly benefited its participants by providing a unique educational experience and an opportunity to interact with modeling teams from around the world. The workshop is expected to have further repercussions on the development of operational atmospheric modeling systems, by giving modeling groups an opportunity to assess and intercompare their models with other advanced modeling systems. Modeling groups have leveraged this information to improve their own models, which will in turn positively impact the quality of weather and climate simulations going forward.

The workshop has advanced our knowledge of (1) the relative behaviors exhibited by atmospheric dynamical cores, (2) differences that arise among mechanisms for coupling the physical parameterizations and dynamical core, and (3) the impacts of variable-resolution refinement regions and transition zones in global atmospheric simulations. Notably, the use of idealized test cases isolating specific phenomena gave us a unique opportunity to assess specific differences that arise due to the choice of dynamical core. A key outcome of the workshop was the development of a standard test case suite and benchmark set of simulations that can be used for assessment of any future dynamical core. The test cases introduced in the 2016 workshop build on the previous DCMIP test case suites (Jablonowski et al., 2008; Ullrich et al., 2012) with tests that now incorporate simplified moist physics.

This paper is the first in a series of papers documenting the results of this workshop. This paper is intended to provide a review of the multitude of technologies and techniques that have been developed for non-hydrostatic global atmospheric modeling and provide an introduction to the intercomparison strategy that has been employed by DCMIP. For easy reference,

a list of mathematical symbols that are employed in this paper (and subsequent DCMIP papers) is given in Table 1. Section 2 provides a brief overview of each of the participating models. The middle of this paper is dedicated to an overview of techniques available for building the infrastructure of a global dynamical core: Section 3 describes model grids that have been employed by modern modeling systems; section B describes different formulations of the moist dynamical equations; section 5 5 describes diffusion, stabilization, filters and fixers employed by these models; and section 6 describes temporal discretizations. Conclusions and discussion then follow in section E.

## 2 Dynamical Cores

This section provide a brief overview of key discretization choices, along with unique features or design specifications from 10 participating dynamical cores. Further details on these choices can be found in subsequent sections. Simulation results and model descriptions have been submitted from ten dynamical cores (see Table 2). Although all models solve the Euler equations (or a close approximation of these equations), essentially every model employs a different formulation with a distinct set of prognostic variables. The derivation of many of these formulations can be found in Appendix B. The prognostic variables employed and horizontal discretizations for these dynamical cores are summarized in Table 3. The vertical coordinate chosen by these models and vertical staggering of variables is summarized in Table 4. Standard options for diffusion, stabilization, 15 filters, or fixers along with the temporal discretization for these models is summarized in Table 5.

### 2.1 Colorado State University Model (CSU)

The CSU model uses an optimized geodesic grid to discretize the sphere, with height as the vertical coordinate. The model is based on the non-hydrostatic Unified System of equations proposed by Arakawa and Konor (2009), which filters vertically propagating sound waves but allows the Lamb wave and does not require a reference state. The horizontal wind field is 20 determined by predicting the vertical component of the vorticity and the divergence of the horizontal wind, and then solving a pair of two-dimensional Poisson equations for a stream function and velocity potential. Horizontal diffusion is included in the form of a fourth-order Laplacian operator applied on constant height surfaces ( $\nabla_z^4$ ) that acts on the vorticity, divergence, potential temperature, and tracer. The CSU model supports both third-order and fifth-order upstream-weighted finite-volume advection schemes, with positivity preservation enforced via mass borrowing.

### 25 2.2 DYNAMICO

DYNAMICO is a mimetic finite-difference / finite-volume model, solving initially the hydrostatic primitive equations and recently extended to solve the shallow-atmosphere fully-compressible equations. DYNAMICO's design uniquely combines a representation of the prognostic and diagnostic fields following the ideas of discrete differential geometry, (Dubos et al., 2015). It includes a novel Hamiltonian formulation of the equations of motion in non-Eulerian coordinates (Dubos and Tort, 30 2014) which is imitated at the discrete level using building blocks from the literature,(Thuburn et al., 2009; Ringler et al., 2010), and (up to the addition of explicit diffusion) leads to an energy-conserving spatial discretization. It also incorporates a

**Table 1.** List of symbols used by DCMIP

Symbol	Description
$\lambda$	Longitude (in radians)
$\varphi$	Latitude (in radians)
$z$	Height with respect to mean sea level (set to zero)
$p_s$	Surface pressure ( $p_s$ of moist air if $q > 0$ )
$\Phi_s$	Surface geopotential
$z_s$	Surface elevation with respect to mean sea level (set to zero)
$u$	Zonal wind
$v$	Meridional wind
$w$	Vertical velocity
$\mathbf{u}$	3D wind vector
$\mathbf{u}_h$	Horizontal wind vector
$\mathbf{v}_h$	Horizontal wind vector with covariant components
$\omega$	Vertical pressure velocity
$D$	Divergence of the horizontal wind vector
$\zeta$	Vertical component of relative vorticity
$p$	Pressure (pressure of moist air if $q > 0$ )
$e$	Internal energy
$\rho$	Total air density
$\rho_d$	Dry air density
$\rho_s$	Pseudo-density
$T$	Temperature
$T_v$	Virtual temperature
$\theta$	Potential temperature
$\theta_v$	Virtual potential temperature
$q$	Specific humidity
$P_{ls}$	Large-scale precipitation rate
$q$	Water vapor mixing ratio
$q_c$	Cloud water mixing ratio
$q_r$	Rain water mixing ratio
$q_i$	General tracer mixing ratio

novel explicit-implicit splitting which results in a simple, efficient and scalable implicit solver while allowing stable time steps close or identical to those of the hydrostatic solver (Dubos and Dubey, in preparation). In addition, it features a conservative positive-definite transport scheme based on a slope-limited finite-volume approach (Dubey et al., 2015).

**Table 2.** Participating modeling centers and associated dynamical cores that have submitted simulation results.

Acronym	Model Name	Modeling Center or Group
CSU	Colorado State University Model	Colorado State University, USA
DYNAMICO	DYNAMICO	Institut Pierre Simon Laplace (IPSL), France
FV <sup>3</sup>	GFDL Finite-Volume Cubed-Sphere Dynamical Core	Geophysical Fluid Dynamics Laboratory, USA
FVM	Finite Volume Module of the Integrated Forecasting System	European Centre for Medium-Range Weather Forecasts
GEM	Global Environmental Multiscale model	Environment and Climate Change Canada
HOMME	High-Order Method Modeling Environment	Sandia National Laboratories / University of Colorado, Boulder
ICON	Icosahedral Non-hydrostatic model	Max-Planck-Institut für Meteorologie, Germany
MPAS	Model for Prediction Across Scales	National Center for Atmospheric Research, USA
NICAM	Non-hydrostatic Icosahedral Atmospheric Model	AORI / JAMSTEC / AICS, Japan
Tempest	Tempest Non-hydrostatic Atmospheric Model	University of California, Davis, USA

**Table 3.** Details on the prognostic variables and horizontal discretization for participating dynamical cores. Equation set indicates whether a model is hydrostatic (H) or nonhydrostatic (NH), and whether the model presently supports the deep-atmosphere formulation (D). Only two numerical methods are represented among participating models, namely finite-volume (FV) and spectral-element (SE). More details on horizontal staggering can be found in section 4.1.

Acronym	Equation set	Prognostic Variables	Horizontal Grid	Numerical method	Horizontal staggering
CSU	NH (Unified)	$\zeta, D, w, p_s, \theta_v, q$	Geodesic (§3.4)	FV	Z-grid
DYNAMICO	H/NH (D)?	$\mathbf{v}_h, \rho_s w, \rho_s, \rho_s \theta_v, \Phi, \rho_s q$	Geodesic (§3.4)	FV	C-grid
FV <sup>3</sup>	NH	$\mathbf{u}_h, w, \rho_s, \rho_s \theta_v, \Phi, \rho_s q$	Cubed-sphere (§3.2)	FV	D-grid
FVM	NH (D)	$\rho_d, \rho_d \mathbf{u}_h, \rho_d w, \rho_d \theta, (\Phi), \rho_d q$	Octahedral reduced (§3.6)	FV	A-grid
GEM	NH	$\mathbf{u}_h, w, \log \partial s / \partial \zeta, \log T_v, q$	Yin-yang (§3.7)	FV	C-grid
HOMME	H	$\mathbf{u}_h, \rho_s, T_v, \rho_s q$	Cubed-sphere (§3.2)	SE	A-grid
ICON	NH	$\mathbf{u}_h, w, \rho, \theta_v, \rho q$	Icosahedral triangular (§3.3)	FV	C-grid
MPAS	NH	$\rho_d \mathbf{u}_h, \rho_d w, \rho_d, \rho_d \theta_v, \rho_d q$	CCVT (§3.5)	FV	C-grid
NICAM	NH	$\rho \mathbf{u}_h, \rho w, \rho, \rho e, \rho q$	Geodesic (§3.4)	FV	A-grid
Tempest	NH	$\mathbf{u}_h, w, \rho, \rho \theta_v, \rho q$	Cubed-sphere (§3.2)	SE	A-grid

**Table 4.** Vertical coordinates and staggering for participating dynamical cores. More details on vertical coordinates can be found in section 4.3, and details on vertical staggerings can be found in section 4.2.

Acronym	Vertical Staggering	Vertical Coordinate
CSU	Lorenz	fixed height
DYNAMICO	Lorenz	floating mass?
FV <sup>3</sup>	Lorenz	floating mass
FVM	Co-located	(floating) height
GEM	Modified Charney-Phillips (§4.2)	log pressure (§4.3.2)
HOMME	Co-located	floating mass
ICON	Lorenz	fixed height
MPAS	Lorenz	fixed height
NICAM	Lorenz	fixed height
Tempest	Lorenz	fixed height

**Table 5.** Standard options for diffusion, stabilization, filters, or fixers in participating dynamical cores, detailed in section 5, and temporal discretization, detailed in section 6.

Acronym	Standard options for diffusion, stabilization, filters, or fixers	Temporal discretization
CSU	4th-order horizontal hyperdiffusion	3rd-order Adams-Bashforth (AB3)
DYNAMICO	?	ARK232 (Giraldo et al., 2013)
FV <sup>3</sup>	Divergence damping, monotonic limiting, hyperviscosity	Forward-backwards (Lin and Rood, 1997)
FVM	Momentum dissipation, scalar diffusion, absorbing layer	Semi-implicit (Smolarkiewicz et al., 2014) (§6.2)
GEM	Hyperviscosity	Semi-implicit (Girard et al., 2014) (§6.3)
HOMME	Hyperdiffusion	KGU53 (Guerra and Ullrich, 2016)
ICON	Divergence damping, Smagorinsky, hyperdiffusion	Predictor-corrector
MPAS	Smagorinsky, hyperdiffusion	Split-explicit (Klemp et al., 2007)
NICAM	3D divergence damping, Smagornisky, hyperviscosity	Split-explicit (Klemp et al., 2007)
Tempest	Hyperdiffusion	ARS232 (Ascher et al., 1997)

### 2.3 FV Cubed (FV<sup>3</sup>)

The GFDL Finite-Volume Cubed-Sphere Dynamical Core (FV<sup>3</sup>, or sometimes written FV3) is a fully finite-volume discretization of the fully-compressible non-hydrostatic Euler equations on the equiangular gnomonic cubed-sphere grid with flow-following Lagrangian vertical coordinate (Lin, 2004). The Lagrangian vertical coordinate deforms so that the flow is

constrained to follow the Lagrangian surfaces, allowing vertical transport to be represented implicitly without additional advection terms (see section 4.3.3 below). Fluxes are computed using the Piecewise-Parabolic Method of Colella and Woodward (1984) with an optional monotonicity constraint; in non-hydrostatic applications the monotonicity constraint is used primarily for tracer transport. The discretization is on the C-D grid as described by Lin and Rood (1997) (see section 4.1.1). Since divergence is effectively invisible to the solver, a divergence damping is applied to control numerical noise as divergent modes cascade to the grid scale. Implicit viscosity is applied through the monotonicity constraint; if non-monotonic advection is used for the momentum and total air mass a weak explicit hyperviscosity is applied for stability and to alleviate numerical noise. Explicit viscosity is applied every acoustic timestep.

The prognostic horizontal winds are stored in the native Gnomonic local coordinate. The non-hydrostatic solver adds a prognostic vertical velocity and geometric height of each grid cell, which can then be used to compute density. All variables are 3D cell-mean values, except for the horizontal winds, which are 2D face-mean values on their respective staggerings; as a result, diagnostic vorticity is a 3D cell-mean value.

## 2.4 Finite-Volume Module (FVM) of the Integrated Forecasting System

The finite-volume module (FVM) of the Integrated Forecasting System (IFS) is presently under development at ECMWF (Smolarkiewicz et al., 2016). FVM solves the fully compressible Euler equations in geospherical coordinates. Both deep-atmosphere and shallow-atmosphere equations are available by means of simple switches. The formulation incorporates a generalized, optionally time-dependent, terrain-following vertical coordinate based on height. A centered two-time-level semi-implicit integration scheme is employed with 3D implicit treatment of acoustic, buoyant, and rotational modes (Smolarkiewicz et al., 2014) (see section 6.2). The associated 3D Helmholtz problem is solved iteratively using a bespoke preconditioned Generalised Conjugate Residual approach. The dynamics time step is adapted at every time step according to a given maximum advective CFL number (typically somewhat smaller than 1). The physics time step is identical to the dynamics time step.

The integration procedure uses the multidimensional flux-form Eulerian non-oscillatory MPDATA advection scheme (Smolarkiewicz and Szmelter, 2005; Kühnlein and Smolarkiewicz, 2017). The horizontal spatial discretization is fully unstructured finite-volume using the median-dual approach. This is combined with a structured-grid finite-difference approach in the vertical direction; see Smolarkiewicz et al. (2016) for an exposition. In both the horizontal and the vertical discretization, all prognostic variables are co-located. The median-dual finite-volume mesh in the horizontal is developed about the points/nodes of the octahedral reduced Gaussian grid (see section 3.6). The octahedral reduced Gaussian grid is also employed in the spectral dynamical core of the current operational IFS at ECMWF, which facilitates interoperability of the two formulations. However, we note that FVM is not restricted to this grid and offers capabilities towards a broad classes of meshes including adaptivity.

No explicit diffusion is applied in FVM for DCMIP, apart from the momentum dissipation and scalar diffusion required for some of the test cases, which is the vertical dissipation/diffusion in the planetary boundary layer parametrization and the constant-coefficient second-order dissipation/diffusion in the supercell test. An absorbing layer in the first latitude ring around the poles is optionally used in the form of a Rayleigh-type forcing to the prognostic variables.

## 2.5 Global Environmental Multiscale (GEM) Model

The GEM model (Girard et al., 2014) is used for operational forecasting at Environment and Climate Change Canada. The horizontal discretization uses the Yin-Yang grid (Kageyama and Sato, 2004), with Arakawa C-grid staggering. The vertical coordinate is a unique hybrid terrain-following coordinate of a log-hydrostatic-pressure type and the vertical discretization is

5 based on the Charney-Phillips grid (see section 4.2). A two time level semi-Lagrangian implicit time discretization is implemented as described in section 6.3. It gives rise to an *iterative* process where each step requires the solution of a linear system of equations that is reduced to a Helmholtz problem for one composite variable. For this problem, a direct solver is involved, using the Schwarz-type domain decomposition method (Qaddouri et al., 2008). The dynamics and physics are time split. To eliminate numerical noise, an explicit hyperviscosity is employed for wind components and tracers via applications of the  
10 Laplacian operator. Viscosity operations are applied after the completion of the physics time step.

## 2.6 High-Order Method Modeling Environment (HOMME)

The High Order Method Modeling Environment (HOMME) (Taylor and Fournier, 2010; Dennis et al., 2012) is currently used by both the Community Atmosphere Model Spectral Element (CAM-SE) dynamical core and Accelerated Climate Model for Energy Atmosphere (ACME-A) dynamical core. As a general framework for several dynamical methods, HOMME includes a

15 hydrostatic option and an experimental non-hydrostatic core, although only shallow-atmosphere mode is available at present. The model is designed to be mass and energy conserving, with nearly optimal parallel scalability at large core counts. The CAM-SE dynamical core is a hydrostatic model that partitions the globe horizontally using an unstructured grid of quadrilateral elements. These elements are arranged in a cubed-sphere structure, although arbitrary quadrilateral grids with conforming edges may also be employed. The model employs a hybrid terrain-following pressure coordinate in the vertical. The fluid  
20 equations are discretized using dimension splitting, with a nodal 4th order spectral element discretization in the horizontal and the mimetic (mass and energy conserving) 2nd order finite difference discretization of Simmons and Burridge (1981) in the vertical. Fields are co-located in the horizontal in the sense that they share the same 4th order basis functions. Lorenz staggering is employed in the vertical with  $(u, v, T_v, p)$  placed on layer midpoints and diagnostic vertical fluxes  $m\dot{\eta}$  are computed on layer interfaces.

## 25 2.7 Icosahedral Non-hydrostatic model (ICON)

The ICON model (Zängl et al., 2015) discretizes the compressible equations for a shallow atmosphere in vector invariant form for the horizontal wind on a triangular Arakawa C-grid and a smoothed terrain-following height-based Lorenz grid. Prognostic horizontal velocities are stored as normal wind components  $v_n$  at the edge mid-points of full levels. Prognostic vertical wind  $w$  is stored at the circumcenters of the triangles on half levels. The discretization in time employs a two time level predictor  
30 corrector scheme, which is explicit in all terms except for those describing the vertical propagation of sound waves. Time splitting is applied between the dynamics that is forced by slow physics on the one hand and horizontal diffusion, tracer transport, and fast physics. One complete time step typically includes 5 dynamical sub-steps. The average air mass flux of

the dynamical sub-steps is provided to the tracer transport to allow for a mass-consistent transport. For stabilization of the divergence term on the triangular C-grid the divergence in a triangle is computed from modified normal wind components resulting from a weighted average including normal winds on edges of adjacent cells. Further divergence damping is applied to the normal wind at every sub-step. Rayleigh damping is applied to the vertical wind in layers close to the model top in  
5 order to avoid the reflection of gravity waves. The horizontal diffusion, which is applied at full model time steps, combines a flow dependent Smagorinsky scheme with a background 4th order Laplacian diffusion operator. For tracer transport a flux form semi-Lagrangian scheme with monotone flux limiters is used, which grants local mass conservation and consistency with the air motion. The numerical methods have been chosen for high numerical efficiency, and they rely on next-neighbor communication only, thus allowing massive parallelization.

## 10 2.8 Model for Prediction Across Scales (MPAS)

The Model for Prediction Across Scales (MPAS) (Skamarock et al., 2012) uses an Arakawa C-grid built on a centroidal Voronoi tessellation (see section 3.5), in conjunction with the mimetic TRiSK discretization (Thuburn et al., 2009; Ringler et al., 2010). Advection terms are nominally third- to fourth-order and are handled in accordance with Skamarock and Gassmann (2011). In the vertical, MPAS employs a Lorenz-type second-order nodal finite volume method with a smoothed terrain-following  
15 height coordinate. The prognostic variables are dry air pseudodensity  $\tilde{\rho}_d$ , dry momentum  $\tilde{\rho}_d \mathbf{u}$ , and a modified moist potential temperature. Integration in time is handled via the split-explicit method of Klemp et al. (2007). Various filters are available for controlling spurious oscillations, including Smagorinsky-type eddy viscosity, and fourth-order hyperdiffusion.

## 2.9 Non-hydrostatic ICosahedral Atmospheric Model (NICAM)

NICAM uses the finite-volume discretization of the fully compressible non-hydrostatic Euler equations on a geodesic grid  
20 optimized by the spring dynamics method proposed by Tomita et al. (2002). The system of equations is transformed to the standard terrain-following height coordinate system (Tomita and Satoh, 2004). Instead of the temperature or the potential temperature, the total energy is chosen as a prognostic variable, following the method of (Satoh, 2002, 2003). All of the prognostic variables are collocated horizontally at the mass centroid of each hexagonal/pentagonal cell to mitigate accuracy reduction under cell averaging, which is required in converting cell integrated quantities to point values at cell centroids. The use  
25 of cell centroids ensures quasi second-order accuracy of the gradient and divergence operators of NICAM (Tomita et al., 2001). The Arakawa C-grid staggering is used vertically. A two-stage Runge-Kutta scheme is usually used to reduce computational burden although a three-stage Runge-Kutta scheme (Wicker and Skamarock, 2002) is recommended. The split-explicit time discretization is used for the horizontally propagating sound waves with the divergence damping term (Skamarock and Klemp,  
1992). The implicit time discretization is adopted for the vertically propagating one. A variant of the piecewise linear transport  
30 scheme (Miura, 2007; Niwa et al., 2011) is used with a flux limiter of Thuburn (1997) for passive tracer transports.

## 2.10 Tempest

The Tempest model (Ullrich, 2014b; Guerra and Ullrich, 2016) is an experimental testbed for high-performance numerical methods that uses a horizontal spectral element discretization and vertical nodal finite volume method with Lorenz staggering based on the cubed-sphere grid with terrain-following height-based coordinate. The implementation includes both fully explicit time integration, and a horizontally-explicit vertically-implicit formulation that is solved with a third-order implicit-explicit additive Runge-Kutta scheme from Ascher et al. (1997).

## 3 Model Grids

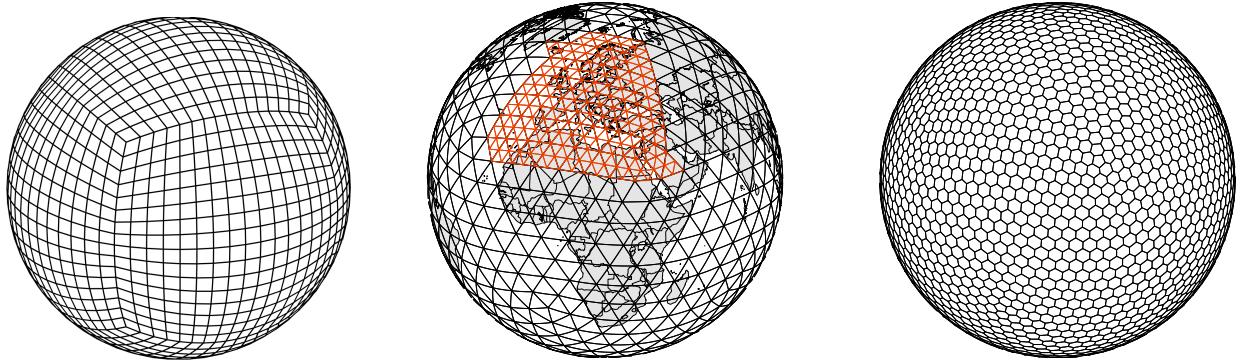
Model grids are a primary concern for dynamical cores, since they determine how degrees of freedom are distributed across the globe (Staniforth and Thuburn, 2012). In order to meet demands for high computational efficiency and equal partitioning of computation across large parallel systems, modern dynamical cores have been exploring a number of options for model grids. The choice of model grid can be motivated by simplicity, as in the case of the latitude-longitude grid, by a desire to maintain a local Cartesian structure, as with the cubed-sphere grid, or to support grid isotropy and homogeneity, as with many of the hexagonal or Voronoi grids that have been employed. The choice of grid may be further decided by the numerical method – for instance, most finite element models rely on grids consisting entirely of quadrilaterals. Inevitably a choice must be made, and the pros and cons of that choice will have repercussions throughout the remainder of the model. To better understand the options that are available to dynamical core developers, we begin by reviewing many of the model grids that have been employed in global dynamical cores around the world.

### 3.1 Latitude-longitude grid

The classic latitude-longitude grid consists of a subdivision of the sphere produced by subdividing along lines of constant latitude and longitude. Because of the convergence of grid lines near the poles, the operational use of this grid requires that the associated numerical scheme be resilient to arbitrarily small Courant number, or that polar filtering be employed to remove unstable computational modes (Lin, 2004). This grid is employed by the UK Met Office (Davies et al., 2005).

### 3.2 Cubed-sphere grid

The equiangular gnomonic cubed-sphere grid (Sadourny, 1972; Ronchi et al., 1996; Putman and Lin, 2007) consists of six Cartesian patches arranged along the faces of a cube which is then inflated onto a spherical shell. More information on this choice of grid can be found in Ullrich (2014b). On the equiangular cubed-sphere grid, coordinates are given as  $(\alpha, \beta, p)$ , with central angles  $\alpha, \beta \in [-\frac{\pi}{4}, \frac{\pi}{4}]$  and panel index  $p$ . The structure of this grid supports refinement through stretching (Schmidt, 1977; Harris et al., 2016) or nesting (Harris and Lin, 2013). The Cartesian structure of cubed-sphere grid panels is advantageous for numerical methods that are formulated in Cartesian coordinates, or that utilize dimension splitting. Nonetheless, special



**Figure 1.** (Left) A cubed-sphere grid with 16 elements along each cubed-sphere edge. (Center) An icosahedral (triangular) grid with additional refinement over Europe, as indicated in red. (Right) An icosahedral (hexagonal) grid.

treatment of the panel boundaries is often necessary since they represent coordinate discontinuities. This grid is depicted in Figure 1 (left).

### 3.3 Icosahedral (triangular) grid

For the common global applications, the icosahedral triangular grid is derived from the spherical icosahedron that consists of 5 20 equilateral spherical triangles, 30 great circle edges and 12 vertices. These initial triangles are then subdivided repeatedly until the desired mean resolution is obtained. For a single subdivision each edge is divided in  $n$  arcs of equal length, thus defining new vertices, which by proper connection to other new vertices result in  $n^2$  triangles filling the original triangle. By construction the new vertices share 6 triangles, thus the refinement process breaks the initial isotropy of the icosahedron and results in non-equilateral triangles of different sizes.

- 10 Several methods are available for subdividing the triangular regions. One such approach is implemented by the ICON grid generator, which allows an “arbitrary” subdivision factor  $n$  for the first refinement step only, the so-called root refinement. Typical choices are  $n = 2, 3$  or  $5$ . All additional  $m$  refinement steps use  $n = 2$ , i.e. are bisection steps. A global grid resulting from a root division factor  $n$  and  $m$  bisections, denominated as  $RnBm$  grid, has  $n_c = 20 \cdot n^2 \cdot 2^{2m}$  cells,  $n_e = 3/2 \cdot n_c$  edges and  $n_v = 10 \cdot n^2 \cdot 2^{2m} + 2$  vertices. The anisotropy of global grids is reduced by the spring dynamics of Tomita et al. (2001).  
15 An example of such a grid is depicted in Figure 1 (center). For a discussion of the effective resolution see Dipankar et al. (2015). The ICON grid generator further allows for inset regional grids, produced by additional refinement steps that are only applied over a limited region, or set of regions. The dynamical core then allows for either one-way or two-way coupling of the refined region to the parent model. The current operational numerical weather prediction of the Deutscher Wetterdienst (German Weather Service, DWD) for instance uses a  $R3B7$  global grid with 2949120 cells and 13 km mean resolution in  
20 combination with a refined region over Europe at 6.5 km resolution.

### 3.4 Icosahedral (hexagonal) grid / geodesic grid

The icosahedral (hexagonal) grid, also commonly referred to as the geodesic grid, is obtained by taking the dual to the icosahedral (triangular grid), namely by replacing grid nodes with spherical polygons. Their cells are hexagonal, except for twelve pentagonal cells. Given a icosahedral-triangular mesh, vertices of the corresponding icosahedral-hexagonal mesh are defined as

- 5 either circumcenters or barycenters of triangles, leading to either a Voronoi mesh, used by DYNAMICO, or a barycentric mesh, used by NICAM. A Voronoi mesh has the property that triangular edges are perpendicular to edges of hexagons/pentagons, facilitating the formulation of certain finite-difference and finite-volume numerical schemes. The resulting highly homogeneous and isotropic grid then appears analogous to the grid in Figure 1 (right). Unlike the cubed-sphere and icosahedral (triangular) grid, grid cells on this the geodesic grid are guaranteed to be edge-neighbors (cells that share a given edge) if they are also
- 10 node-neighbors (cells that share a given node).

It is useful to optimize icosahedral-hexagonal grids as well. DYNAMICO applies a number of iterations of Lloyds algorithm (Lloyd, 1982), following which one replaces the vertices of the original triangular mesh by the centroid of hexagons/pentagons, then re-generates the icosahedral-hexagonal mesh. This improves the homogeneity of the grid (e.g. ratio of largest cell area to smallest cell area) but several thousand iterations may be required for a significant improvement.

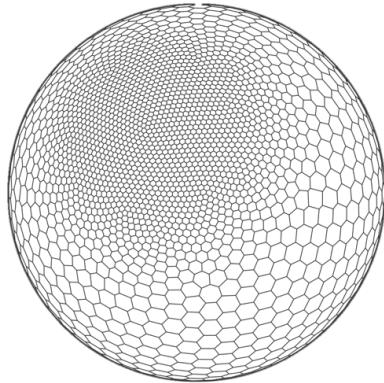
### 15 3.5 Constrained Centroidal Voronoi Tessellation (CCVT) grids

Given a set of  $N$  distinct points on the sphere  $x_i$  (referred to as the generators,  $1 \leq i \leq N$ ), the *Voronoi tessellation* (or the *Voronoi diagram*) associated with the generators is the set of polygons  $\Omega_i$  consisting of all points that are closer (in the sense of great-circle distance) to  $x_i$  than any other  $x_j$  with  $i \neq j$  (Okabe et al., 2009). For a given set of generators, this tiling is unique and completely covers the sphere, and so can be employed in conjunction with many finite volume methods. However,

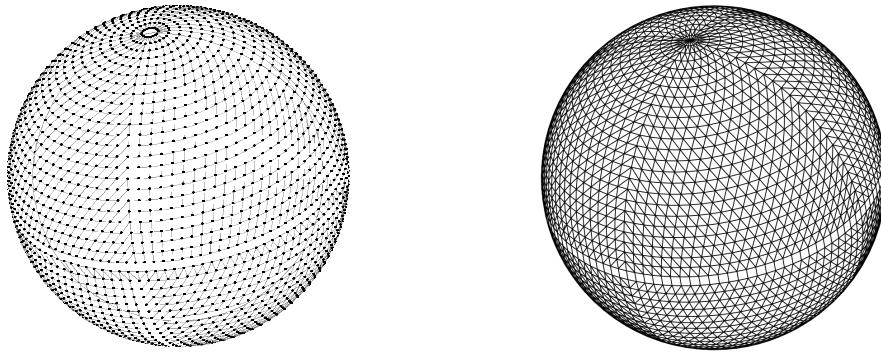
- 20 for an arbitrary set of generators it is easy to produce highly distorted polygons, particularly if the density of generators varies substantially. This has led to the development of *constrained centroidal Voronoi tessellation (CCVT)* (Du et al., 2003), which imposes the additional requirement that the set of generators be coincident with the centroids of each polygon. Given a desired polygonal density function, several algorithms have been developed to generate CCVTs both in Cartesian and spherical geometry (i.e. for ocean basins or ice sheets) (Ringler et al., 2008). Figure 2 depicts one such CCVT grid that is compatible with
- 25 the MPAS model. CCVT grids are often confused with deformations of the icosahedral (hexagonal) grid described in section 3.4, since both typically contain a large number of hexagonal elements. However, CCVT grids are fundamentally constructed using a very different technique and will often contain polygonal elements with more than six sides.

### 3.6 Octahedral reduced Gaussian grid

As with the classical reduced Gaussian grid of Hortal and Simmons (1991), the octahedral reduced Gaussian grid (Malarde et al., 2016; Smolarkiewicz et al., 2016) specifies the latitudes according to the roots of the Legendre polynomials. The two grids differ in the arrangement of the points along the latitudes, which follows a simple rule for the octahedral grid: starting with 20 points on the first latitude around the poles, four points are added with every latitude towards the equator, whereby the



**Figure 2.** A constrained centroidal Voronoi tessellation grid with localized grid density that could be employed in the MPAS model.

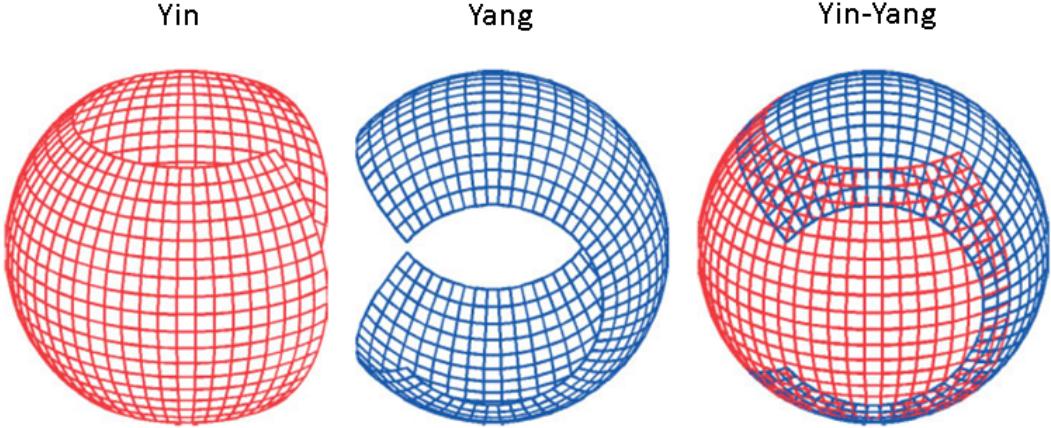


**Figure 3.** Locations of the octahedral reduced Gaussian grid nodes (left), and the edges of the primary mesh connecting the nodes as applied with the finite-volume discretisation in FVM (right). A coarse grid with only 24 latitudes between pole and equator is used for illustration. The dual mesh resolution of the octahedral reduced Gaussian grid is about a factor 2 finer at the poles than the equator; see Smolarkiewicz et al. (2016).

spacing between points along the latitudes is uniform and there are no points at the equator. The octahedral reduced Gaussian grid is suitable for transformations involving spherical harmonics, and has been introduced for global medium-range numerical weather prediction with the spectral dynamical core of the IFS at ECMWF in 2016. Figure 3 depicts the octahedral reduced Gaussian grid nodes together with the edges of the primary mesh as applied in the context of the finite-volume discretisation of FVM (Section 2.4).

### 3.7 Yin-Yang grid

The overset Yin-Yang grid (Kageyama and Sato, 2004) has two Cartesian grid components (subsets of a latitude-longitude grid) which are geometrically identical (see Figure 4). These components are combined to cover a spherical surface with



**Figure 4.** Yin-Yang grid

partial overlap along their borders. The Yin component covers the latitude-longitude region

$$(-\frac{\pi}{4} - \delta_\theta \leq \theta \leq \frac{\pi}{4} + \delta_\theta) \cap (-\frac{3\pi}{4} - \delta_\lambda \leq \lambda \leq \frac{3\pi}{4} + \delta_\lambda), \quad (1)$$

where  $\delta_\lambda, \delta_\theta$  are small buffers that are proportional to the respective grid-spacings and are required to enforce a minimum overlap in the overset methodology. For instance, a common configuration employed by the GEM model for DCMIP fixes

- 5  $\delta_\theta = 2$  degrees and  $\delta_\lambda = 3\delta_\theta$ . The Yang component covers an analogous area, but is rotated perpendicularly so as to cover the region of the sphere outside of the Yin grid. This grid is employed by the GEM model, utilizing a pair of regional climate models on the two Cartesian patches.

## 4 Variable staggering and vertical coordinates

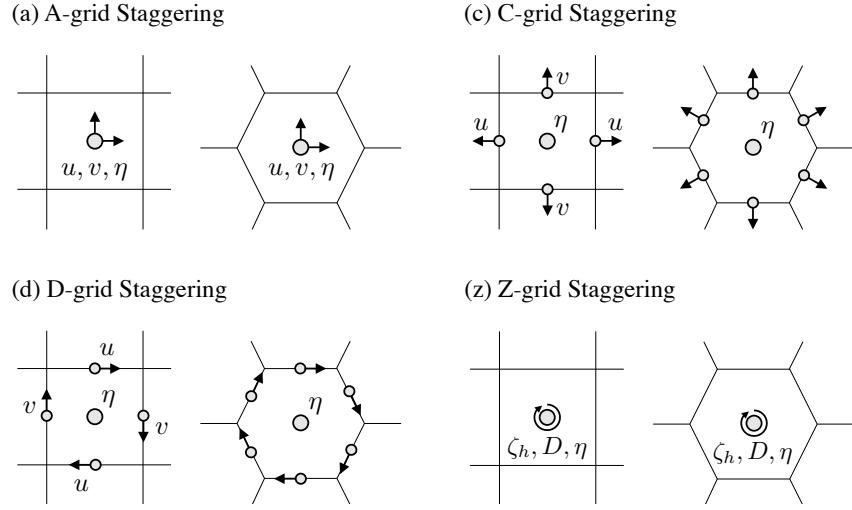
Although the horizontal grid determines the approximate density and distribution of grid points, the placement of prognostic

- 10 (and diagnostic) variables within each grid volume is still free to be specified. Hence, in this section we turn our attention to the placement of prognostic variables on the grid and the choice of vertical coordinate.

### 4.1 Horizontal staggering

The horizontal placement of variables impacts a number of properties of the numerical method, including the ability of the model to conserve invariant such as energy and enstrophy, any computational modes that might arise due to differencing, and

- 15 the dispersion properties of the method (Randall, 1994), and the maximum stable timestep size (Ullrich, 2014a). The original four Arakawa grids (Arakawa and Lamb, 1977), denoted with letters A- through D- were initially designed for rectilinear meshes, but were later adapted for a variety of unstructured grids. Later other grid types were added, including the Z-grid, which used the vertical component of vorticity and the horizontal divergence in place of the velocity components (Randall,



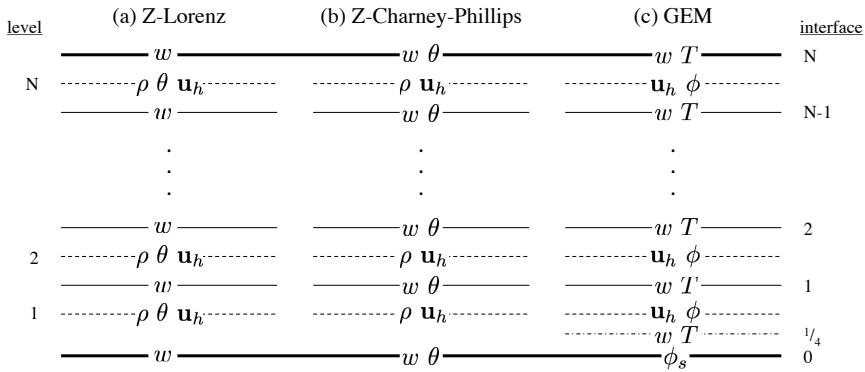
**Figure 5.** Horizontal staggering options represented among DCMIP models, in this case depicted on a rectilinear grid and geodesic grid.

1994), and the ZM-grid, which extends the B-grid to hexagons by placing the velocity at hexagonal nodes (Ringler and Randall, 2002). New staggerings are now being developed in the context of finite element methods, using mixed finite element methods (Cotter and Shipton, 2012). Among the models that participated in DCMIP, only four grids were represented: The A-grid, which involves co-location of all velocity components; the C-grid, which places perpendicular velocity components on grid edges; the D-grid, which places parallel velocity components on grid edges; and the Z-grid (see Figure 5).

There are arguments to be made for and against each of these grids: the A-grid is relatively simple to implement and tends to support large time step sizes, but also supports a  $2\Delta x$  computational mode (even in the context of finite element methods); the C-grid has excellent dispersion properties, and does not support extraneous computational modes when the number of edges is equal to twice the number of volumes, but also has imposes a reduced maximum timestep size relative to the A-grid; the D-grid provides a better representation of vorticity, but suffers to similar computational modes as the A-grid; finally, the Z-grid yields optimal dispersion properties, but requires the inversion of a Helmholtz problem at each timestep to extract the velocity field from the divergence and vorticity.

#### 4.1.1 C-D-Grid in the FV<sup>3</sup> model

Although velocities in the FV<sup>3</sup> model are stored in accordance with the D-grid arrangement, at the intermediate stages of the forward-backwards timestepping scheme, velocities are actually prognosed on the C-grid. The intermediate velocities then act as a simplified Riemann solver: the intermediate stage velocities are time-centered and can be used to compute the fluxes and advance the flux terms by a full acoustic timestep. More details on this approach can be found in Lin and Rood (1997).



**Figure 6.** (a) A Lorenz-type variable staggering for a model utilizing height coordinates, (b) a Charney-Phillips-type variable staggering for a model utilizing height coordinates, (c) a modified Charney-Phillips-type staggering used in the GEM model that introduces a new near-surface level for vertical velocity and temperature.

## 4.2 Vertical staggering

Along with the choice of prognostic variables, the vertical discretization of the equations of motion also allows for the staggered placement of prognostic variables. As with hydrostatic models, certain discretizations give rise to spurious computational modes that can contaminate the solution (Tokioka, 1978; Arakawa and Moorthi, 1988). The choice of vertical staggering may 5 also impact many physically-relevant properties of the model near the grid scale, such as the phase speed of Rossby waves (Thuburn and Woolings, 2005). Finally, the choice of vertical staggering can have impacts on the physics-dynamics coupling Holdaway et al. (2013a, b). Taken altogether, these issues suggest care should be taken when selecting the discretization. For 10 this reason, co-located discretizations, where all degrees of freedom are geometrically co-located, are not frequently adopted in global atmospheric models. In particular, it is common to employ either: (a) a Lorenz-type staggering (Lorenz, 1960), which or (b) a Charney-Phillips-type staggering (Charney and Phillips, 1953), which places horizontal velocity and buoyancy variables 15 on model levels and vertical velocity and thermodynamic variables on model interfaces (see Figure 6). These approaches can be further augmented as needed, for instance by shifting the vertical velocity and thermodynamic variables from the bottom boundary to an intermediate level, as in the GEM model. Note that, in general, tracer variables are co-located with the buoyancy variable.

## 4.3 Vertical Coordinates

Most atmospheric models use dimension splitting in order to separate the evolution equations for momentum into horizontal and vertical directions. Whereas “horizontal” in this context either refers to either contravariant velocities, which are perpendicular to the vertical, or covariant velocities, which are directed along coordinate (e.g. terrain-following) surfaces, the vertical is

strictly aligned with the radial vector pointing from the center of the Earth. In this context, the use of dimension splitting allows us to choose an arbitrary vertical coordinate  $s$  with the only requirement being that it is a monotonic function of the altitude  $z$ . Typically  $s$  is chosen so that the Earth's surface (the bottom boundary of the atmosphere) is a coordinate surface, allowing for easy specification of boundary conditions for the prognostic equations – this leads to the so-called “terrain-following” family of vertical coordinates. Perhaps the most common terrain-following coordinate is from Gal-Chen and Somerville (1975), which is in terms of the altitude  $z$  and takes the form

$$s(\mathbf{x}, z) = z_{\text{top}} \left[ \frac{z - z_s(\mathbf{x})}{z_{\text{top}} - z_s(\mathbf{x})} \right], \quad (2)$$

where  $\mathbf{x}$  denotes the horizontal position,  $z_s(\mathbf{x})$  is the height of the topography at that position, and  $z_{\text{top}}$  denotes the height of the model top (typically independent of position). Analogous formulations are available for mass-based ( $\sigma$ -coordinates) and entropy-based vertical coordinates. Because the sharp variations in the coordinate surfaces are preserved far above a rough lower-boundary, new coordinate formulations have been proposed that smooth coordinate surfaces, such as Schär et al. (2002) or Klemp (2011). All models in this paper use some variant of terrain-following coordinates, although work on cut-cell, embedded boundary and immersed boundary representations is ongoing (e.g. Lock et al. (2012)). Note that time-dependent vertical coordinates are allowed and are typically referred to as “floating” coordinates. Several examples of vertical coordinates are now given.

#### 4.3.1 Mass-based coordinates

Mass-based coordinates are a generalization of pressure-based coordinates to non-hydrostatic models, with a vertical coordinate defined as the total gravity-weighted overhead mass,

$$s = \int_z^{\infty} \rho g dz. \quad (3)$$

Under this definition,

$$\frac{\partial s}{\partial z} = -\rho g. \quad (4)$$

The mass-based coordinate gives rise to a prognostic equation for hydrostatic surface pressure through vertical integration of (B55),

$$\frac{\partial}{\partial t} \int \rho_s ds + \int \nabla_s \cdot (\rho_s \mathbf{u}_h) ds = 0, \quad (5)$$

where the integrals are taken over the column.

#### 4.3.2 GEM zeta coordinate

The vertical coordinate of GEM model, denoted  $\zeta$ , is a hybrid terrain-following coordinate of a log-hydrostatic-pressure type. Taking  $s$  (denoted  $\pi$  in GEM documentation) as given in (3), then  $\zeta$  is given by the relation

$$\log s = A(\zeta) + B(\zeta) [\log s(z_s) - \zeta_s], \quad (6)$$

with  $A(\zeta) = \zeta$ ,  $B(\zeta) = \left(\frac{\zeta - \zeta_{top}}{\zeta_s - \zeta_{top}}\right)^r$  and where  $\zeta_s = \log(10^5)$ ,  $\zeta_{top} = \log(s_{top})$ , and  $r$  is a variable exponent providing added freedom for adjusting the thickness of model layers over high terrain.

#### 4.3.3 Floating Lagrangian coordinates (DYNAMICO, FV<sup>3</sup> and HOMME)

In the floating Lagrangian formulation (Lin, 2004) the vertical coordinate is chosen to represent an artificial tracer with monotonically increasing or decreasing mixing ratio  $s$  in the vertical. The actual mixing ratio at initiation is arbitrary, and can be constructed to be height-like (i.e.,  $s = z$ ), or mass-like, i.e.

$$s = \int_z^\infty \rho_0 g dz, \quad (7)$$

in which case a 3D reference density field  $\rho_0$  can be imposed. Of primary importance is the fact that the vertical coordinate satisfy

$$10 \quad \dot{s} = \frac{ds}{dt} = 0, \quad (8)$$

which greatly simplifies the associated prognostic velocity and continuity equations. Floating Lagrangian coordinates are often paired with a vertical remapping operation that corrects for strong grid distortions that may occur after sufficiently long model integrations.

### 4.4 Prognostic equations

15 The Naviér-Stokes equations that govern atmospheric motion can take on many forms, depending on the choice of prognostic variables and coordinate system. A derivation of many forms of these equations can be found in Appendix B. The particular equations used by the model, and the discrete form of these equations, can impact many important aspects of the model, including the presence of computational modes, the accuracy of the model in representing the physical modes of the atmosphere (Thuburn and Woollings, 2005), and the ability of the model to conserve important invariants such as energy (Dubos and Tort, 2014). The remainder of this section gives some specific examples of prognostic equations used by the DCMIP models.

#### 4.4.1 DYNAMICO

DYNAMICO is a shallow-atmosphere fully-compressible model based on a mass-based vertical coordinate and a Hamiltonian formulation (Dubos and Tort, 2014). Prognostic variables are pseudo-density  $\rho_s$ , mass-weighted tracers (potential temperature, water species), geopotential  $\Phi$ , “horizontal” covariant components of momentum and mass-weighted vertical momentum  $W = \rho_s g^{-2} d\Phi/dt = \rho_s g^{-1} w$ . Prognostic equations are in flux-form for mass, tracers and  $W$ , in advective form for  $\Phi$  and in vector-invariant form for “horizontal” momentum.

#### 4.4.2 FV<sup>3</sup>

The hydrostatic FV<sup>3</sup> model uses a mass-based floating Lagrangian coordinate along with the shallow-atmosphere approximation (Lin, 2004). Prognostic equations include horizontal velocity in 2D vector-invariant form (B38), pseudo-density (B55), and virtual potential temperature (B57). The non-hydrostatic model further incorporates prognostic geopotential (B27) and  
5 vertical momentum (B37).

#### 4.4.3 FVM

FVM solves conservation laws for dry mass (9a), momentum (9b), and dry entropy (9c) in Eulerian flux-form, which are similar to (B9) for  $\rho_d$ , (B23), and (B13) for  $\theta$ , respectively. Underlying the governing equations in FVM is a perturbational form with respect to a balanced ambient state, and a generalized curvilinear coordinate formulation in a geospherical framework. These  
10 governing equations can concisely be written as (Smolarkiewicz et al., 2017a)

$$\frac{\partial \mathcal{G}\rho_d}{\partial t} + \bar{\nabla} \cdot (\mathbf{v}\mathcal{G}\rho_d) = 0, \quad (9a)$$

$$\frac{\partial \mathcal{G}\rho_d \mathbf{u}}{\partial t} + \bar{\nabla} \cdot (\mathbf{v}\mathcal{G}\rho_d \mathbf{u}) = \mathcal{G}\rho_d \left( -\theta_\rho \tilde{\mathbf{G}} \bar{\nabla} \phi' - \mathbf{k} \frac{g}{\theta_a} \left( \theta' + \theta_a \left( \frac{q'_v}{\varepsilon} - q'_v - q_c - q_r \right) \right) - 2\boldsymbol{\Omega} \times \left( \mathbf{u} - \frac{\theta_\rho}{\theta_{\rho a}} \mathbf{u}_a \right) + \mathcal{M}' \right), \quad (9b)$$

$$\frac{\partial \mathcal{G}\rho_d \theta'}{\partial t} + \bar{\nabla} \cdot (\mathbf{v}\mathcal{G}\rho_d \theta') = -\mathcal{G}\rho_d \tilde{\mathbf{G}}^T \mathbf{u} \cdot \bar{\nabla} \theta_a, \quad (9c)$$

$$\phi' = c_{pd} \left[ \left( \frac{R_d}{p_0} \rho_d \theta (1 + q_v / \varepsilon) \right)^{R_d / c_{vd}} - \pi_a \right]. \quad (9d)$$

15 Dependent variables in (9) are dry density  $\rho_d$ , 3D physical velocity vector  $\mathbf{u}$ , potential temperature perturbation  $\theta'$ , and a modified Exner pressure perturbation  $\phi'$ , with the thermodynamic variables related by the gas law (9d). Primes indicate perturbations with respect to the prescribed ambient state denoted by subscript "a"; see also Prusa et al. (2008); Smolarkiewicz et al.  
20 (2014) for discussion. In the shallow- versus deep-atmosphere form of (9), gravity is constant  $g \equiv g_c$  or varies with height as  $g = g_c (a/(a+z))^2$ , respectively. As far as geometric aspects are concerned, the nabla operator  $\bar{\nabla}$  represents the 3D vector of partial derivatives with respect to the curvilinear coordinates, along with the Jacobian  $\mathcal{G}$ , a matrix of metric coefficients  $\tilde{\mathbf{G}}$ , its transpose  $\tilde{\mathbf{G}}^T$ , and the contravariant velocity  $\mathbf{v} = \tilde{\mathbf{G}}^T \mathbf{u}$ ; see also Prusa and Smolarkiewicz (2003) and Kühnlein et al. (2012). The symbol  $\mathcal{M}'(\mathbf{u}, \mathbf{u}_a, \theta_\rho / \theta_{\rho a})$  in (9b) subsumes spherical Christoffel symbols (Smolarkiewicz et al., 2017a).

#### 4.4.4 GEM

In GEM the non-hydrostatic equations are written explicitly as deviations from hydrostatic balance represented by

$$25 \quad \mu = \frac{\partial p}{\partial s} - 1, \quad (10)$$

where  $s$  (denoted  $\pi$  in GEM documentation) is given by (3) and in which case the equations of GEM model (Girard et al., 2014) are:

$$\frac{d\mathbf{u}_h}{dt} + f\mathbf{k} \times \mathbf{u}_h + R_d T_v \nabla_\zeta \log p + (1 + \mu) \nabla_\zeta \Phi = 0, \quad (11)$$

$$\frac{dw}{dt} - g\mu = 0, \quad (12)$$

$$5 \quad \frac{d}{dt} \log \left( \frac{\partial s}{\partial \zeta} \right) + \nabla_\zeta \cdot \mathbf{u}_h + \frac{\partial \dot{\zeta}}{\partial \zeta} = 0, \quad (13)$$

$$\frac{d \log T_v}{dt} - \frac{R_d}{c_p} \frac{d \log p}{dt} = 0, \quad (14)$$

$$\frac{\partial \Phi}{\partial s} + \frac{R_d T_v}{p} = 0, \quad (15)$$

$$\frac{d\Phi}{dt} - gw = 0. \quad (16)$$

#### 4.4.5 HOMME

10 HOMME uses a floating Lagrange formulation (section 4.3.3) for tracer advection and optionally for the vertical dynamics. The 2D vector invariant form of the prognostic horizontal velocity equations (B62) is employed, in conjunction with prognostic temperature (B16) and surface pressure (5).

In addition to the primitive equation model, HOMME also hosts an experimental non-hydrostatic dynamical core which retains many of the features of the traditional CAM-SE model. This model makes use of the shallow-atmosphere Euler equations, 15 as first described by Laprise (1992). In its current incarnation, unstructured spectral elements are retained in the horizontal with matching spectral elements in the vertical. Prognostic variables include the 3d wind velocity components  $u, v, w$  [Equation?], the surface pressure  $p_s$  (5), potential temperature (B12), and geopotential (B26).

#### 4.4.6 ICON

ICON is based on Gassmann and Herzog (2008) using terrain-following  $z$ -coordinates with prognostic density (B10), virtual 20 potential temperature (B57), and vector-invariant form for horizontal momentum in vector-invariant form (B33) and vertical momentum (B32). The specific heat capacities and ideal gas constant are approximated to be equal to their dry values  $R^* \approx R_d$ ,  $c_p^* \approx c_{pd}$  and  $c_v^* \approx c_{vd}$ . The model also uses a prognostic equation for Exner pressure to simplify vertical sound wave propagation, given by

$$\frac{\partial \pi}{\partial t} + \frac{R_d}{c_{vd}} \frac{\pi}{\rho \theta_v} \nabla \cdot (\mathbf{u}_h \rho \theta_v) = \hat{Q}, \quad (17)$$

25 where  $\hat{Q}$  is an appropriately formulated diabatic heat term. The horizontal uses a Arakawa C-grid formulation on the triangular grid to prognose horizontal velocities normal to triangle edges  $v_n$ , making use of reconstructed tangential velocity components  $v_t$ .

#### 4.4.7 Tempest

Tempest is a shallow-atmosphere Eulerian model with terrain-following  $z$ -coordinates with prognostic density (B55), virtual potential temperature (B57), and vector-invariant form for covariant horizontal velocity (B76) and vertical momentum (B32).

## 5 Diffusion, Stabilization, Filters and Fixers

- 5 Most dynamical cores implement specialized techniques for diffusion or stabilization (see Table 5). Diffusion is a numerical technique that removes spurious numerical noise from the simulation, where the numerical noise typically arises because of inaccuracies in the treatment of waves with wavelengths near the grid scale. Stabilization is a numerical technique that prevents energy growth and allows the model to be run over long periods. Diffusion or stabilization options include physically-motivated turbulence parameterizations, added viscosity or hyperviscosity terms with tunable coefficients, off-centering, or wave-mode  
10 filters. Since the discretization can also lead to an unphysical loss of mass or energy, mass or energy fixers are also employed to replace lost mass or energy to the system. A comprehensive overview of schemes for diffusion and stabilization schemes can be found in Jablonowski and Williamson (2011). In this section we discuss some of the diffusion and stabilization strategies employed by the DCMIP suite of dynamical cores.

### 5.1 HOMME / Tempest

- 15 In both HOMME and Tempest, scalar hyperviscosity is employed for  $\rho$ ,  $\theta$  and tracer variables via repeated application of a scalar Laplacian (Ullrich, 2014b). Vector hyperviscosity is also applied by decomposing the horizontal vector Laplacian into divergence damping and vorticity damping terms via the vector identity

$$\nabla^2 \mathbf{u}_h = \nabla \nabla \cdot \mathbf{u}_h + \nabla \times \nabla \times \mathbf{u}_h. \quad (18)$$

- Both viscosity operations are applied after the completion of all Runge-Kutta sub-cycles. Several limiter options are available  
20 for tracer transport including a sign-preserving limiter and a monotone optimization base limiter described in Guba et al. (2014).

### 5.2 ICON

- The ICON model employs damping and diffusion operators for numerical stabilization and dynamic closure. The details of this scheme appear in sections 2.4 and 2.5 of Zängl (2014), and are summarized here. For damping, in the corrector step a  
25 fourth-order divergence damping term  $F_d(\mathbf{v})$  is applied in order to allow calling the computationally more expensive diffusion operator (see below) at the physics time steps without incurring numerical stability problems under extreme conditions.

$$F_d(\mathbf{v}) = -f_d \overline{a_c}^2 \nabla \tilde{\nabla} \cdot \left( \nabla \left( \tilde{\nabla} \cdot \mathbf{v} + \frac{\Delta}{\Delta z} \left( w - \overline{w_{cc}}^{c_i} \right) \right) \right). \quad (19)$$

$f_d$  typically attains values between  $\frac{1}{1000\Delta t}$  and  $\frac{1}{250\Delta t}$ , and  $\overline{a_c}$  is the global mean cell area.

ICON also includes Rayleigh damping on  $w$  following Klemp et al. (2008), which serves to prevent unphysical reflections of gravity waves at the model top. The Rayleigh damping is restricted to a fixed number of levels below the model top, and the damping coefficient is given by a hyperbolic tangent.

The horizontal diffusion consists of a flow dependent second-order Smagorinsky diffusion of velocity ( $F_{D2}(v_n)$ ) and potential temperature ( $F_{D2}(\theta)$ ) combined with a fourth-order background diffusion of velocity  $F_{D4}(v_n)$ , defined via

$$F_{D2}(v_n) = 4K_h \tilde{\nabla}^2(v_n), \quad F_{D2}(\theta) = a_c \tilde{\nabla} \cdot \left( K_h \frac{\Delta\theta}{\Delta n} \right). \quad F_{D4}(v_n) = -k_4 a_e^2 \tilde{\nabla}^2(\tilde{\nabla}^2(v_n)). \quad (20)$$

An empirically determined offset of  $0.75k_4a_e$  is subtracted from  $K_h$  in order to avoid excessive diffusion under weakly disturbed conditions.

A fourth-order computational diffusion is also available for vertical wind speed  $w$ . This filter term is needed at resolutions of O(1 km) or finer because the advection of vertical wind speed has no implicit damping of small-scale structures. This term appears as

$$F_D(w) = -k_w a_c^2 \nabla^2(\nabla^2(w)) \quad (21)$$

### 5.3 FV<sup>3</sup>

Explicit dissipation in FV<sup>3</sup> is applied separately to the divergence and to the horizontal fluxes in the governing equations. The D-grid discretization applies no direct implicit dissipation to the divergence, so divergence damping is an intrinsic part of the solver algorithm since otherwise there are no processes by which energy contained in the divergent modes is removed at the grid scale. FV<sup>3</sup> has options for fourth-, sixth-, or eighth-order divergence damping; a second-order option is also available for use in idealized convergence tests, which can be applied in addition to the higher-order diffusion. The monotonicity constraint used in computing the fluxes in the momentum, thermodynamic, and mass continuity equations is sufficient to damp and stabilize the non-divergent component of the flow. If additional damping is desired, or if the non-monotonic advection is used, there is an option to apply hyperdiffusion to the fluxes in each of these equations, with the exception of the tracer transport, which always uses monotonic transport with no explicit diffusion. The hyperdiffusion is of the same order as but much smaller than the divergence damping. Both divergence damping and hyperdiffusion are applied along the Lagrangian surfaces and are re-computed every acoustic timestep.

Wave absorption at the model top is also provided by the flexible-lid (constant-pressure) upper boundary; FV<sup>3</sup> also applies second-order diffusion to all fields, except the tracers, to create a sponge layer, typically comprising the top two layers of the domain, to damp other signals reaching the top of the domain. An energy-conserving Rayleigh damping, applied consistently to all three components of the winds, is also available, which is strongest in the top layer of the domain and becomes weaker with distance until reaching a runtime-specified cut-off pressure.

FV<sup>3</sup> has an option to restore lost energy by the adiabatic dynamics, in whole or a fraction thereof (decided by a namelist option at runtime), by globally adding a Exner-function weighted potential temperature increment. This is only done before the physics is called and is not used in idealized simulations.

## 5.4 GEM

An explicit hyperviscosity in GEM is handled via applications of the Laplacian operator for both wind components and tracers. A vertical sponge layer, which uses a Laplacian operator, is employed on wind components and  $T_v$  with a vertical modulation on the topmost levels. For stabilization purpose, the temporal discretization of GEM also uses an off-centering parameter.

- 5 The quasi-monotone semi-Lagrangian (QMSL) method (Bermejo and Staniforth, 1992) is used operationally to ensure tracer monotonicity for specific humidity and different hydrometeors. Other options are now available in GEM including a mass conserving monotonic scheme (Sørenson et al., 2013) and a global mass fixer (Bermejo and Conde, 2002). Those approaches have been evaluated using chemical constituents such as ozone (de Grandpré et al., 2016).

## 6 Temporal Discretizations

- 10 A wide variety of temporal discretizations available among global atmospheric models, particularly to address issues related to the large horizontal-vertical aspect ratio typical in these systems. In this section we briefly discuss several modern approaches that are presently employed. [reword?]

### 6.1 Mixed Implicit-Explicit, Forward-Backward, Semi-Implicit and Additive Runge-Kutta Schemes

- Implicit-explicit schemes are a broad category of time integration schemes that divide the terms of the prognostic equations 15 into a set of explicitly integrated terms and implicitly integrated terms. At the very least, terms associated with vertically propagating sound waves are included among the implicit terms. For the remaining terms, there is some freedom in choosing how to integrate terms associated with vertical advection and horizontally propagating sound waves. Semi-implicit schemes are one such class of schemes that typically incorporate horizontally propagating sound waves into the implicit solve, and so rely on a global Helmholtz-type solve. Additive Runge-Kutta schemes are another mechanism to ensure high-order temporal 20 accuracy, and many such schemes have been described throughout the literature (see, for example, Weller et al. (2013); Ullrich and Jablonowski (2012b)). Several examples of these schemes can be found among the DCMIP models:

- FV<sup>3</sup> and its predecessors are integrated using a forward-backwards integration for the Lagrangian dynamics. With the exception of the pressure-gradient force, all of the terms in the momentum, energy, and mass equations are expressable as fluxes, and so can be integrated using the explicit forward-in-time algorithm described by Lin and Rood (1997). The horizontal component 25 of the pressure-gradient force is evaluated backwards-in-time using the algorithm of Lin (1997); the non-hydrostatic component of the vertical pressure gradient force is evaluated using a semi-implicit solver. This forward-backward timestep is referred to as the “acoustic” timestep, although the full solver is advanced on each of these acoustic timesteps. Physics tendencies are applied impulsively at prescribed intervals, consistent with the forward-in-time discretization; the physics timestep is typically much longer than the acoustic timestep.
- 30 DYNAMICO uses an additive Runge-Kutta time scheme with two Butcher tableaus, one explicit and one implicit. A Hamiltonian splitting decides which terms of the equations of motion are treated explicitly or implicitly (Dubos and Dubey, in

preparation). As a result the implicit terms couple the vertical acceleration due to the pressure gradient and the adiabatic pressure change due to vertical displacements of fluid parcels. The resulting implicit problem reduces to independent, scalar, purely vertical, nonlinear problems which are solved to machine precision in two Newton iterations involving one tridiagonal solve each. The overall time scheme has a HEVI (horizontally explicit, vertically implicit) structure. Currently the second-order 5 3-stage scheme ARK(2,3,2) is used (Giraldo et al., 2013).

ICON consists of a two-time-level predictor corrector scheme, which is explicit for all terms except for those describing the vertical propagation of sound waves. No time splitting is used with respect to sound waves, because the ratio of the maximum wind speed in the mesosphere, which is in part covered by the vertical domain, can be close to one. Instead time splitting is employed to dynamics on the one hand and horizontal diffusion, tracer transport, fast physics on the other hand. Typically a 10 full time step consists of 4 or 5 dynamical sub-steps in which a constant forcing originating from the slow physics is applied. Mass-consistent transport is achieved by passing time-averaged air-mass fluxes from the dynamical sub-steps to the transport scheme. The details of the predictor corrector scheme, including measures to increase the numerical efficiency and to optimize the accuracy, are described in section 2.4 of Zängl (2014).

MPAS uses a split-explicit formulation (Klemp et al., 2007) consisting of an outer Runge-Kutta loop (typically RK3) and 15 inner acoustic loop. At the beginning of each Runge-Kutta sub-cycle, tendencies are computed for each of the prognostic variables and stored for the duration of the sub-cycle. Several iterations of an acoustic loop are then performed with a time-step much smaller than require for the Runge-Kutta sub-cycle. Within the acoustic loop, an implicit solve for vertically-integrated sound waves is performed to avoid timestep constraints that may arise from vertically-propagating sound waves.

Tempest uses the ARS(2,3,2) scheme described in Ascher et al. (1997), with all horizontal and vertical advection terms 20 treated explicitly and the remaining vertical terms, associated with sound wave propagation, treated implicitly.

## 6.2 The FVM Semi-Implicit method

A characteristic feature of the FVM (Section 2.4) time-stepping scheme is the 3D implicit treatment of the fast buoyant and acoustic modes, and the slow rotational modes. Therefore, the model time step is identical for all processes and selected according to the stability of the advective transport scheme—i.e. the time step is continuously adapted according to a given maximum 25 advective Courant number permitted by the MPDATA scheme. A comprehensive discussion of the integration scheme can be found in Smolarkiewicz et al. (2014, 2016); Kühnlein and Smolarkiewicz (2017) for dry dynamics, and in Kurowski et al. (2014); Smolarkiewicz et al. (2017a) for moist-precipitating dynamics. Here, we provide an outline of the basic integration scheme for the compressible Euler equations (9). The scheme employs the two-time-level second-order-accurate template algorithm given as

$$30 \quad \psi_{\mathbf{i}}^{n+1} = \mathcal{A}_{\mathbf{i}}(\tilde{\psi}^n, \mathbf{V}^{n+1/2}, (\mathcal{G}\rho_d)^n, (\mathcal{G}\rho_d)^{n+1}) + 0.5 \Delta t R^{\psi}|_{\mathbf{i}}^{n+1}, \quad \tilde{\psi}^n \equiv \psi^n + 0.5 \Delta t R^{\psi}|^n. \quad (22)$$

where  $\psi$  represents the solution variable,  $R^{\psi}$  is the respective rhs,  $\mathcal{A}$  symbolises the advective transport operator—the non-oscillatory finite-volume MPDATA bespoke for the compressible dynamics on the hybrid horizontally-unstructured vertically-

structured mesh (Kühnlein and Smolarkiewicz, 2017)—and the vector index  $\mathbf{i} \equiv (k, i)$  denotes the spatial position on the computational mesh.

The integration of the system (9) can basically be divided into three steps. First, the homogenous mass continuity equation is integrated with  $\psi \equiv \rho_d$ ,  $\mathbf{V} \equiv \mathbf{v}\mathcal{G}$ , and  $R^{\rho_d} \equiv 0$  in (22). Second, the thermodynamic (9c), momentum (9b), and moisture equations enter (22) with  $\psi = u, v, w, \theta', \dots$ ,  $\mathbf{V} \equiv \mathbf{v}\mathcal{G}\rho_d$ , and the rhs  $R^\psi$  which is generally depending on all prognostic variables.  
5 The high degree of implicitness in the representation of the rhs forcings is achieved by inverting the overall discrete system (22) to obtain closed-form expressions for the velocity updates—the procedure is facilitated by the co-located arrangement of all variables on the computational mesh. Retained on the rhs of the derived closed-form velocity expressions is the pressure gradient term. The subsequent third step in the solution procedure is to formulate an implicit boundary value problem  
10 for the pressure variable  $\phi'$  using an advective form of the equation of state (9d). An  $\mathcal{O}(\Delta t^2)$  integration of this equation with a Euler backward scheme, in the spirit of (22), leads a Helmholtz equation (Smolarkiewicz et al., 2014). The associated 3D elliptic boundary value problem is solved iteratively using a preconditioned Generalised Conjugate Residual approach; see Smolarkiewicz and Szmelter (2011) and references therein. Nonlinear terms in  $R^\psi|^{n+1}$  and the solution-dependent coefficients of the Helmholtz equation are lagged behind and executed in an outer iteration.

### 15 6.3 A semi-Lagrangian implicit time discretization in the GEM model

GEM differs from the approaches above by using a semi-Lagrangian advection. Any model equations, prognostic or diagnostic, is written in the form

$$\frac{dF}{dt} + G = 0, \quad (23)$$

where  $\frac{d}{dt}$  is the Lagrangian derivative,  $F$  containing the terms subject to this operator,  $G$  the remaining terms. The semi-  
20 Lagrangian approach consists in the following space-time discretization of (23)

$$\frac{F^A - F^D}{\Delta t} + \left( \frac{1}{2} + \epsilon \right) G^A + \left( \frac{1}{2} - \epsilon \right) G^D = 0, \quad (24)$$

where  $A$  stands for the arrival position at model grid point  $(\mathbf{r}_h, \zeta, t)$  and  $D$  for the departure position  $(\mathbf{r}_h - \Delta\mathbf{r}_h, \zeta - \Delta\zeta, t - \Delta t)$  due to the displacements  $\Delta\mathbf{r}_h, \Delta\zeta$  having occurred during the timestep  $\Delta t$ .  $G$  is averaged between these two positions with a possible slight off-centering  $\epsilon$ . The displacements are themselves calculated solving, again using the Lagrangian method, the  
25 equations:

$$\frac{d\mathbf{r}_h}{dt} - \mathbf{u}_h = 0; \frac{d\zeta}{dt} - \dot{\zeta} = 0, \quad (25)$$

discretized in the same way (trapezoidal method) though without off-centering:

$$\frac{\Delta\mathbf{r}_h}{\Delta t} - \frac{\mathbf{u}_h^A + \mathbf{u}_h^D}{2} = 0; \frac{\Delta\zeta}{\Delta t} - \frac{\dot{\zeta}^A + \dot{\zeta}^D}{2} = 0. \quad (26)$$

The process is of course a *multi-step iterative* one since both positions and velocities at departure positions (past time  $t - \Delta t$ ) are unknown as well as, of course, the velocities at arrival positions (time  $t$ ). Once a first estimate of the departure positions

is obtained, the model equations are solved to obtain a first estimate of the velocities at time  $t$ . The model equations must be solved simultaneously and this is only possible for the linear part  $L$  which becomes a *matrix inversion problem*. Hence a suitable linearization is considered. The unknown (arrival) linear  $L$  and non-linear  $N$  parts are then separated from the known  
5 (first departure estimate) remaining  $R$  part. Thus, first separating space-times, (24) is rewritten as follows

$$\frac{F^A}{\tau} + G^A = \frac{F^D}{\tau} - \beta G^D \equiv R^D, \quad (27)$$

where  $\tau = (1/2 + \epsilon) \Delta t$  and  $\beta = (1/2 - \epsilon) / (1/2 + \epsilon)$ . Second separating linear from non-linear parts, we get:

$$L^A + N^A = R^D, \quad (28)$$

with

$$10 \quad L^A = \left[ \frac{F^A}{\tau} + G^A \right]_{linear}, \quad and \quad N^A \equiv \frac{F^A}{\tau} + G^A - \left[ \frac{F^A}{\tau} + G^A \right]_{linear}. \quad (29)$$

Note that both  $F$  and  $G$  may require linearization.  $L^A$  may then be obtained if  $N^A$  is first guessed: Once  $L^A$  is found, an estimate of  $N^A$  is obtained and  $L^A$  is recalculated. This is called the *non-linear iteration process* (one iteration is usually sufficient). The overall process is then repeated once starting from a new estimate of the departure positions.

There are two intensive calculation sections in this process: the so-called semi-Lagrangian calculations (twice estimating  
15 departure positions, twice interpolating right-hand sides  $R$  on departure positions), and solution of the linear system (four times). Each time, the linear system is reduced to a Helmholtz problem for one composite variable. For this problem, a direct solver is involved, using the Schwarz-type domain decomposition method on a Yin-Yang grid (Qaddouri et al., 2008). The composite variable solution is then used to update the prognostic variables (back substitution). At the end of the time-step,  
20 the static halo region of both panels of the Yin-Yang grid is updated (Qaddouri and Lee, 2011). All required interpolations throughout the semi-Lagrangian process and between Yin and Yang grids are cubic interpolations.

## Appendix A: Notation

This section provides an overview of the shared notation adopted by DCMIP for describing the equations and discretizations that have been adopted by the dynamical cores involved in this project. Table 1 lists the symbols used in this paper. A list of physical constants which are used throughout this document is given in Table 6. Great circle distance is used throughout the  
25 project and is computed via

$$R_c(\lambda_1, \varphi_1; \lambda_2, \varphi_2) = a \arccos (\sin \varphi_1 \sin \varphi_2 + \cos \varphi_1 \cos \varphi_2 \cos(\lambda_1 - \lambda_2)). \quad (A1)$$

**Table 6.** A list of physical constants used in this document.

Constant	Description	Value
$a_{\text{ref}}$	Radius of the Earth	$6.37122 \times 10^6 \text{ m}$
$\Omega_{\text{ref}}$	Rotational speed of the Earth	$7.292 \times 10^{-5} \text{ s}^{-1}$
$g_c$	Gravitational acceleration	$9.80616 \text{ m s}^{-2}$
$p_0$	Reference pressure	1000 hPa
$c_{pd}$	Specific heat capacity of dry air at constant pressure	$1004.5 \text{ J kg}^{-1} \text{ K}^{-1}$
$c_{pw}$	Specific heat capacity of water vapor at constant pressure	$1930.0 \text{ J kg}^{-1} \text{ K}^{-1}$
$c_{vd}$	Specific heat capacity of dry air at constant volume	$717.5 \text{ J kg}^{-1} \text{ K}^{-1}$
$c_{vw}$	Specific heat capacity of water vapor at constant volume	$1460.0 \text{ J kg}^{-1} \text{ K}^{-1}$
$R_d$	Gas constant for dry air	$287.0 \text{ J kg}^{-1} \text{ K}^{-1}$
$R_w$	Gas constant for water vapor	$461.5 \text{ J kg}^{-1} \text{ K}^{-1}$
$\varepsilon$	Ratio of $R_d$ to $R_w$	0.622
$M_v$	Constant for virtual temperature conversion	0.608
$\rho_{\text{water}}$	Reference density of water	$1000 \text{ kg m}^{-3}$

## Appendix B: Moist Nonhydrostatic Equation Sets

In this appendix we provide a detailed derivation of the fluid equations utilized by nonhydrostatic models. The material derivative is used for quantities in the Lagrangian frame (following individual air parcels), and is given by

$$\frac{d}{dt} = \frac{\partial}{\partial t} + \mathbf{u} \cdot \nabla, \quad (\text{B1})$$

where  $\mathbf{u}$  denotes the 3D vector velocity. Note that tracer variables  $q_i$ , including specific humidity  $q$ , satisfy the simple Lagrangian relationship

$$5 \quad \frac{dq_i}{dt} = 0. \quad (\text{B2})$$

### B1 Diagnostic relationships

The atmospheric fluid is assumed to be an ideal gas. For moist air, the ideal gas constant  $R^*$ , specific heat capacity at constant pressure  $c_p^*$  and specific heat capacity at constant volume  $c_v^*$  are given by

$$R^* = R_d + (R_w - R_d)q, \quad c_p^* = c_{pd} + (c_{pw} - c_{pd})q, \quad c_v^* = c_{vd} + (c_{vw} - c_{vd})q. \quad (\text{B3})$$

- 10 Note that in many models,  $R^*$ ,  $c_p^*$  and  $c_v^*$  are approximated by  $R_d$ ,  $c_{pd}$  and  $c_{vd}$ , respectively. Dry air, water vapor and moist air quantities all satisfy the linear relationship  $R = c_p - c_v$ . For a two-fluid system (dry air plus water vapor), two independent variables plus the specific humidity  $q$  are needed to describe the thermodynamic state of the system. Key thermodynamic variables include dry air density  $\rho_d$ , moist density  $\rho$ , pressure  $p$ , vapor pressure  $e$ , temperature  $T$ , virtual temperature  $T_v$ ,

Exner pressure  $\pi$ , potential temperature  $\theta$ , and virtual potential temperature  $\theta_v$ . Common ratios  $\kappa = R^*/c_p^*$ ,  $\varepsilon = R_d/R_w$ , and

**15**  $\gamma = c_p^*/c_v^*$  are adopted here.

Relationships between key thermodynamic variables arise from the ideal gas law, along with definitions of Exner pressure, potential temperature and virtual potential temperature

$$p = \rho R_d T_v, \quad \pi = \left( \frac{p}{p_0} \right)^\kappa, \quad \theta = T \left( \frac{p_0}{p} \right)^\kappa, \quad \theta_v = T_v \left( \frac{p_0}{p} \right)^\kappa, \quad (\text{B4})$$

which further give rise to

$$p = \left( \frac{\rho R_d \theta_v}{p_0^\kappa} \right)^\gamma, \quad \pi = \left( \frac{\rho R_d \theta_v}{p_0} \right)^{R^*/c_v^*}, \quad \theta = \frac{T}{\pi}, \quad \theta_v = \frac{T_v}{\pi}. \quad (\text{B5})$$

**5** Note that virtual temperature is typically approximated by

$$T_v \approx T \left( 1 + \frac{(1 - \varepsilon)}{\varepsilon} q \right), \quad (\text{B6})$$

which arises from the exact relationship

$$T_v = \frac{T}{1 - \frac{\varepsilon}{p}(1 - \varepsilon)}, \quad (\text{B7})$$

upon applying  $e/p = q/\varepsilon$  and using a Taylor expansion around  $q = 0$ .

## 10 B2 Prognostic equations for thermodynamic variables

Note that, as a consequence of (B2), the following simplifications can be applied:

$$\frac{1}{T_v} \frac{dT_v}{dt} = \frac{1}{T} \frac{dT}{dt}, \quad \frac{dR^*}{dt} = 0, \quad \frac{dc_p^*}{dt} = 0, \quad \frac{dc_v^*}{dt} = 0. \quad (\text{B8})$$

Mass conservation is typically represented through the continuity equation, which can be written in the Lagrangian frame as

$$\frac{d\rho}{dt} = -\rho \nabla \cdot \mathbf{u}, \quad (\text{B9})$$

**15** or equivalently in the Eulerian frame,

$$\frac{\partial \rho}{\partial t} = -\nabla \cdot (\rho \mathbf{u}). \quad (\text{B10})$$

Further prognostic relationships can be derived from the thermodynamic equation, including the diabatic heating rate  $J$ ,

$$\frac{1}{T} \frac{dT}{dt} - \frac{\kappa}{p} \frac{dp}{dt} = \frac{J}{c_p^*}, \quad (\text{B11})$$

which can be alternatively written as

$$\frac{d\theta}{dt} = \frac{J\theta}{c_p^*}, \quad \text{or} \quad \frac{d\theta_v}{dt} = \frac{J\theta_v}{c_p^*}. \quad (\text{B12})$$

These equations can then be combined with (B9) to obtain

$$\frac{\partial}{\partial t}(\rho\theta_v) + \nabla \cdot (\rho\theta_v \mathbf{u}) = \frac{J\rho\theta_v}{c_p^*}, \quad (\text{B13})$$

or similarly for  $\theta$ . In conjunction with the material derivative of the ideal gas law,

$$\frac{1}{p} \frac{dp}{dt} = \frac{1}{\rho} \frac{d\rho}{dt} + \frac{1}{T_v} \frac{dT_v}{dt}, \quad (\text{B14})$$

the thermodynamic equation can be written in the form

$$\frac{c_v^*}{R^* T_v} \frac{dT_v}{dt} - \frac{1}{\rho} \frac{d\rho}{dt} = \frac{J}{R^*}. \quad (\text{B15})$$

- 5 Then substituting (B9) gives a prognostic equation for virtual temperature,

$$\frac{c_v^*}{R^*} \frac{dT_v}{dt} + T_v \nabla \cdot \mathbf{u} = \frac{JT_v}{R^*}. \quad (\text{B16})$$

The prognostic equation for temperature is identical except with  $T$  substituted for  $T_v$ . An analogous equations for pressure can be obtained through a similar procedure,

$$\frac{c_v^*}{c_p^*} \frac{dp}{dt} + p \nabla \cdot \mathbf{u} = \frac{Jp}{c_p^*}. \quad (\text{B17})$$

- 10 And similarly for Exner pressure,

$$\frac{c_v^*}{R^*} \frac{d\pi}{dt} + \pi \nabla \cdot \mathbf{u} = \frac{J\pi}{c_p^*}. \quad (\text{B18})$$

### B3 Momentum Equations

In coordinate-invariant form the prognostic velocity equations may be written in either the Lagrangian or Eulerian frame as

$$\frac{d\mathbf{u}}{dt} = \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\frac{1}{\rho} \nabla p - 2\boldsymbol{\Omega} \times \mathbf{u} - \nabla \Phi, \quad (\text{B19})$$

- 15 where  $\boldsymbol{\Omega}$  denotes the planetary vorticity vector and  $\Phi$  is the geopotential function. The three terms on the right-hand-side of this expression correspond to pressure gradient, Coriolis, and gravitational force, respectively. In Eulerian form one must be careful with the treatment of the momentum advection term  $\mathbf{u} \cdot \nabla \mathbf{u}$ , since in an arbitrary coordinate frame this term will give rise to Christoffel symbols associated with derivatives of the vector basis. Note that it is common to rewrite the pressure gradient force using the relationship

$$20 -\frac{1}{\rho} \nabla p = -\frac{R_d c_p^*}{R^*} \theta_v \nabla \pi, \quad (\text{B20})$$

which follows from (B4). A second form of (B19) emerges on substituting the vector calculus identity

$$\mathbf{u} \cdot \nabla \mathbf{u} = \nabla K + \boldsymbol{\zeta} \times \mathbf{u}, \quad (\text{B21})$$

where  $K = \frac{1}{2}(\mathbf{u} \cdot \mathbf{u})$  is the 3D Kinetic energy and  $\zeta = \nabla \times \mathbf{u}$  is the 3D relative vorticity vector. This gives rise to the 3D vector-invariant form,

$$25 \quad \frac{\partial \mathbf{u}}{\partial t} = -\frac{1}{\rho} \nabla p - \nabla(K + \Phi) - (\zeta + 2\Omega) \times \mathbf{u}. \quad (\text{B22})$$

Because no gradients of vectors appear in this equation, it avoids derivatives of the coordinate basis that would arise from the momentum transport term  $\mathbf{u} \cdot \nabla \mathbf{u}$  in (B19) (note that these derivatives sometimes appear as Christoffel symbols). In conjunction with (B9), both (B19) also give rise to the flux-form momentum equations,

$$\frac{\partial}{\partial t}(\rho \mathbf{u}) = -\nabla \cdot (\mathbf{u} \otimes \mathbf{u} + \mathcal{I}p) - 2\Omega \times (\rho \mathbf{u}) - \rho \nabla \Phi, \quad (\text{B23})$$

where  $\mathbf{u} \otimes \mathbf{u}$  denotes the outer product and  $\mathcal{I}$  is the identity matrix.

The equations above still provide some flexibility with regards to the choice of  $\Phi$  and  $\Omega$ . For *deep atmosphere* models, one  
5 typically chooses

$$\Phi = ga^2 \left[ \frac{1}{a} - \frac{1}{a+z} \right], \quad \text{and} \quad \Omega = \Omega(\mathbf{k} \sin \varphi + \mathbf{j} \cos \varphi), \quad (\text{B24})$$

where  $g$  is gravitational acceleration at the surface,  $a$  is the radius of the planet,  $\Omega$  is the rotation rate (in  $s^{-1}$ ),  $\varphi$  is the latitude,  $\mathbf{j}$  is the unit vector oriented in the meridional direction, and  $\mathbf{k}$  is the unit vector oriented in the vertical direction. For models that don't utilize a height-based vertical coordinate, the geopotential is generally treated as a prognostic variable, with an evolution  
10 equation that emerges from the definition  $w = dz/dt$ ,

$$\frac{d\Phi}{dt} = \frac{a^2 gw}{(a+z)^2}. \quad (\text{B25})$$

For *shallow atmosphere* models, the geopotential takes the simpler form

$$\Phi = gz, \quad \text{and} \quad \Omega = \Omega \sin \varphi \mathbf{k}, \quad (\text{B26})$$

where  $z$  is the altitude above the surface. In this case we write  $2\Omega = f\mathbf{k}$ , where  $f = 2\Omega \sin \varphi$  is the Coriolis parameter. The  
15 evolution equation for the shallow atmosphere geopotential is then

$$\frac{d\Phi}{dt} = gw. \quad (\text{B27})$$

#### B4 Orthogonal Formulation

Under the orthogonal formulation, projection of a vector field  $\mathbf{b}$  onto its horizontal components is defined via

$$[\mathbf{b}]_z = \mathbf{b} - (\mathbf{b} \cdot \mathbf{k})\mathbf{k}. \quad (\text{B28})$$

20 When applied to the velocity vector this gives rise to the decomposition

$$\mathbf{u} = \mathbf{u}_h + w\mathbf{k}, \quad (\text{B29})$$

where  $\mathbf{k} = \nabla z$  is the unit vector in the vertical direction and  $\mathbf{u}_h = [\mathbf{u}]_z$  ( $\mathbf{u}_h$  is aligned with surfaces of constant  $z$ ). In the orthogonal formulation, the material derivative expands as

$$\frac{d}{dt} = \frac{\partial}{\partial t} + \mathbf{u}_h \cdot \nabla + w \mathbf{k} \cdot \nabla. \quad (\text{B30})$$

- 25 For the special case of the material derivative applied to scalars, this equation can also be written as

$$\frac{d}{dt} = \frac{\partial}{\partial t} + \mathbf{u}_h \cdot \nabla_z + w \frac{\partial}{\partial z}. \quad (\text{B31})$$

where  $\nabla_z b = [\nabla b]_z$  denotes the gradient along constant  $z$  surfaces. From here, the vector-invariant form velocity equation obtained by taking (B22)  $\cdot \mathbf{k}$  expands as

$$\frac{\partial w}{\partial t} = -\frac{1}{\rho} \frac{\partial p}{\partial z} - \frac{\partial}{\partial z} (K + \Phi) - [(\zeta + 2\Omega) \times \mathbf{u}] \cdot \mathbf{k}, \quad (\text{B32})$$

which, from  $\mathbf{u}_h = \mathbf{u} - w\mathbf{k}$ , then gives rise to

$$\frac{\partial \mathbf{u}_h}{\partial t} = -\frac{1}{\rho} \nabla_z p - \nabla_z (K + \Phi) - [(\zeta + 2\Omega) \times \mathbf{u}]_z. \quad (\text{B33})$$

- 5 Due to its association with hydrostatic models, it is common to use the 2D Kinetic energy,  $K_2 = \frac{1}{2}(\mathbf{u}_h \cdot \mathbf{u}_h)$ . Decomposing the momentum transport term into horizontal and vertical components gives

$$\mathbf{u} \cdot \nabla \mathbf{u} = \mathbf{u}_h \cdot \nabla \mathbf{u}_h + (\nabla \times \mathbf{u}_h) \times (w\mathbf{k}) + (\mathbf{u} \cdot \nabla w)\mathbf{k}. \quad (\text{B34})$$

The first term in this expression admits the relationships

$$[\mathbf{u}_h \cdot \nabla \mathbf{u}_h]_z = \nabla_z K_2 + \zeta_h \mathbf{k} \times \mathbf{u}_h, \quad (\text{B35})$$

$$10 \quad (\mathbf{u}_h \cdot \nabla \mathbf{u}_h) \cdot \mathbf{k} = -\mathbf{u}_h \cdot (\mathbf{u}_h \cdot \nabla \mathbf{k}) = -K_2(\nabla \cdot \mathbf{k}) - \frac{1}{2}[\mathbf{u}_h \cdot (\nabla \times \mathbf{u}_t) + (\nabla \times \mathbf{u}_h) \cdot \mathbf{u}_t] \quad (\text{B36})$$

where  $\zeta_h = (\nabla \times \mathbf{u}) \cdot \mathbf{k} = (\nabla \times \mathbf{u}_h) \cdot \mathbf{k}$  is the relative vorticity scalar and  $\mathbf{u}_t = \mathbf{k} \times \mathbf{u}_h$ . Note that this equation does incorporate metric terms associated with horizontal advection of  $\mathbf{k}$  which must be accounted for.

Thus the vertical velocity equation, obtained by taking (B19) $\cdot \mathbf{k}$ , is

$$\frac{\partial w}{\partial t} = \mathbf{u}_h \cdot (\mathbf{u}_h \cdot \nabla \mathbf{k}) - w \frac{\partial w}{\partial z} - \mathbf{u}_h \cdot \nabla_z w - \frac{\partial \Phi}{\partial z} - \frac{1}{\rho} \frac{\partial p}{\partial z} - (2\Omega \times \mathbf{u}_h) \cdot \mathbf{k}. \quad (\text{B37})$$

- 15 Then subtracting (B37) $\cdot \mathbf{k}$  from (B19) gives

$$\frac{\partial \mathbf{u}_h}{\partial t} = -w(\mathbf{u}_h \cdot \nabla \mathbf{k}) - w \frac{\partial \mathbf{u}_h}{\partial z} - \nabla_z(K_2 + \Phi) - \frac{1}{\rho} \nabla_z p - \zeta_h \mathbf{k} \times \mathbf{u}_h - [2\Omega \times \mathbf{u}]_z. \quad (\text{B38})$$

Note that under the shallow atmosphere approximation, the metric term  $\mathbf{u}_h \cdot (\mathbf{u}_h \cdot \nabla \mathbf{k})$  in (B37) is set equal to zero in accordance with Phillips (1966).

## B5 Arbitrary vertical coordinates

- 20 The dynamical equations are now formulated in terms of the vertical coordinate  $s(t, \mathbf{x}, z)$  with  $\partial s / \partial z \neq 0$  everywhere, i.e. following Kasahara (1974) (hereafter K74). Since  $\mathbf{x}$  and  $t$  are shared between the two coordinate systems, the chain rule can be applied to obtain expressions

$$\frac{\partial}{\partial z} = \frac{\partial s}{\partial z} \frac{\partial}{\partial s}, \quad \nabla_s = \nabla_z + (\nabla_s z)(\mathbf{k} \cdot \nabla), \quad \left( \frac{\partial}{\partial t} \right)_s = \frac{\partial}{\partial t} + \left( \frac{\partial z}{\partial t} \right)_s (\mathbf{k} \cdot \nabla), \quad (\text{B39})$$

- which correspond to derivatives in the vertical, in the horizontal and in time. This final expression is used to describe the rate 25 of change of a quantity on  $s$  surfaces. These operators then yield the useful identities

$$\frac{\partial s}{\partial z} = \left( \frac{\partial z}{\partial s} \right)^{-1}, \quad \nabla_z s = - \left( \frac{\partial s}{\partial z} \right) \nabla_s z, \quad \frac{\partial s}{\partial t} = - \frac{\partial s}{\partial z} \left( \frac{\partial z}{\partial t} \right)_s. \quad (\text{B40})$$

From here (B39) also gives rise to

$$\nabla_z = \nabla_s - \frac{\partial s}{\partial z} (\nabla_s z) \frac{\partial}{\partial s}, \quad (\text{B41})$$

- which can be used directly to rewrite (B33) or (B38) in terms of derivatives over  $s$ . Note that the operators  $\nabla_z$  and  $\nabla_s$  are usually introduced in the context of 2D flows, however the construction described here has the advantage of working seamlessly 5 in a 3D context, while admitting the properties  $\mathbf{k} \cdot \nabla_z A = 0$  and  $\mathbf{k} \cdot \nabla_s A = 0$  for any scalar field  $A$ .

From (B39), it can be shown that the 2D divergence on  $s$  surfaces (given by K74 eq. (3.17)) is

$$\nabla_s \cdot \mathbf{u}_h = \nabla_z \cdot \mathbf{u}_h + \left( \frac{\partial s}{\partial z} \right) (\nabla_s z) \cdot \left( \frac{\partial \mathbf{u}_h}{\partial s} \right), \quad (\text{B42})$$

and that the 2D curl is given by

$$\nabla_s \times \mathbf{u}_h = \nabla_z \times \mathbf{u}_h + \left( \frac{\partial s}{\partial z} \right) (\nabla_s z) \times \left( \frac{\partial \mathbf{u}_h}{\partial s} \right), \quad (\text{B43})$$

- 10 where  $\nabla_z \times \mathbf{u}_h = \mathbf{k}(\mathbf{k} \cdot (\nabla \times \mathbf{u}_h))$ . Notably, these expressions are valid for both shallow- and deep-atmosphere formulations.

The generalized velocity  $\dot{s}$  following a fluid parcel is defined by

$$\dot{s} \equiv \frac{ds}{dt} = \frac{\partial s}{\partial t} + \mathbf{u} \cdot \nabla s = \mathbf{u}_h \cdot \nabla_z s + \left[ w - \left( \frac{\partial z}{\partial t} \right)_s \right] \frac{\partial s}{\partial z}. \quad (\text{B44})$$

Then using (B39) and (B44) to rewrite (B31) gives an expression for the material derivative for scalars on  $s$  surfaces,

$$\frac{dA}{dt} = \left( \frac{\partial A}{\partial t} \right)_s - \frac{\partial s}{\partial z} \left( \frac{\partial z}{\partial t} \right)_s \frac{\partial A}{\partial s} + \mathbf{u}_h \cdot \left[ \nabla_s A + (\nabla_z s) \frac{\partial A}{\partial s} \right] + w \frac{\partial s}{\partial z} \frac{\partial A}{\partial s} \quad (\text{B45})$$

$$15 = \left( \frac{\partial A}{\partial t} \right)_s + \mathbf{u}_h \cdot \nabla_s A + \dot{s} \frac{\partial A}{\partial s} \quad (\text{B46})$$

A similar expression arises for vectors, although in this case  $\mathbf{u}_h \cdot \nabla \mathbf{a} \neq \mathbf{u}_h \cdot \nabla_z \mathbf{a}$  implies we cannot use the operator  $\nabla_s$  in the form (B39), and instead obtain

$$\frac{d\mathbf{a}}{dt} = \left( \frac{\partial \mathbf{a}}{\partial t} \right)_s + [\mathbf{u}_h \cdot \nabla \mathbf{a} + (\mathbf{u}_h \cdot \nabla_s z)(\mathbf{k} \cdot \nabla \mathbf{a})] + \dot{s} \frac{\partial \mathbf{a}}{\partial s}. \quad (\text{B47})$$

## B6 Conservation Laws in Arbitrary Vertical Coordinates

20 Using (B42), we observe that the 3D divergence on the sphere takes the form

$$\nabla \cdot \mathbf{u} = \nabla_z \cdot \mathbf{u}_h + \frac{1}{\alpha} \frac{\partial}{\partial z} (\alpha w), \quad (\text{B48})$$

where  $\alpha = 1$  for shallow-atmosphere models and  $\alpha = r^2 = (a + z)^2$  for deep-atmosphere models. Using  $w = dz/dt$ , this last term also takes the form

$$\frac{1}{\alpha} \frac{\partial}{\partial z} (\alpha w) = \frac{\partial w}{\partial z} + \frac{w}{\alpha} \frac{\partial \alpha}{\partial z} = \frac{\partial w}{\partial z} + \frac{1}{\alpha} \frac{d\alpha}{dt}. \quad (\text{B49})$$

Using (B40) to rewrite (B44) gives rise to

$$w = \left( \frac{\partial z}{\partial t} \right)_s + \mathbf{u}_h \cdot \nabla_s z + \dot{s} \left( \frac{\partial s}{\partial z} \right)^{-1}, \quad (\text{B50})$$

which is then differentiated to yield K74 eq. (3.16),

$$\frac{\partial w}{\partial z} = \left( \frac{\partial s}{\partial z} \right) \left[ \frac{d}{dt} \left( \frac{\partial s}{\partial z} \right)^{-1} + \left( \frac{\partial \mathbf{u}_h}{\partial s} \right) \cdot (\nabla_s z) \right] + \frac{\partial \dot{s}}{\partial s} = 0. \quad (\text{B51})$$

5 Substituting this expression into the continuity equation (B9), and using (B48), (B49), and (B51) then leads to

$$\frac{d}{dt} \left[ \alpha \left( \frac{\partial s}{\partial z} \right)^{-1} \rho \right] + \alpha \left( \frac{\partial s}{\partial z} \right)^{-1} \rho \left[ \nabla_z \cdot \mathbf{u}_h + \left( \frac{\partial s}{\partial z} \right) \left( \frac{\partial \mathbf{u}_h}{\partial s} \right) \cdot (\nabla_s z) \right] + \alpha \left( \frac{\partial s}{\partial z} \right)^{-1} \rho \frac{\partial \dot{s}}{\partial s} = 0. \quad (\text{B52})$$

Defining the *pseudodensity* as

$$\rho_s = \alpha \left( \frac{\partial s}{\partial z} \right)^{-1} \rho, \quad (\text{B53})$$

and using (B46) in the form

$$10 \quad \frac{d\rho_s}{dt} = \left( \frac{\partial \rho_s}{\partial t} \right)_s + \mathbf{u}_h \cdot \nabla_s \rho_s + \dot{s} \frac{\partial \rho_s}{\partial s}, \quad (\text{B54})$$

along with (B42) leads to

$$\left( \frac{\partial \rho_s}{\partial t} \right)_s + \nabla_s \cdot (\rho_s \mathbf{u}_h) + \frac{\partial}{\partial s} (\rho_s \dot{s}) = 0. \quad (\text{B55})$$

Hence for any quantity that is conserved following a fluid parcel (*i.e.*,  $dq/dt = 0$ ),

$$\left( \frac{\partial \rho_s q}{\partial t} \right)_s + \nabla_s \cdot (\rho_s q \mathbf{u}_h) + \frac{\partial}{\partial s} (\rho_s q \dot{s}) = 0. \quad (\text{B56})$$

15 In particular, the prognostic equation for virtual potential temperature (or equivalently for potential temperature) reads

$$\left( \frac{\partial \rho_s \theta_v}{\partial t} \right)_s + \nabla_s \cdot (\rho_s \theta_v \mathbf{u}_h) + \frac{\partial}{\partial s} (\rho_s \theta_v \dot{s}) = \frac{J \rho_s \theta_v}{c_p^*}. \quad (\text{B57})$$

## B7 2D Vector Invariant Form

The prognostic equations utilizing horizontal kinetic energy  $K_2$  in place of  $K$  are derived by applying (B39) to (B37), yielding

$$\left(\frac{\partial w}{\partial t}\right)_s = \mathbf{u}_h \cdot (\mathbf{u}_h \cdot \nabla \mathbf{k}) - \mathbf{u}_h \cdot \nabla w + \left(\frac{\partial s}{\partial z}\right) \left\{ \left[ \left(\frac{\partial z}{\partial t}\right)_s - w \right] \frac{\partial w}{\partial s} - \frac{\partial \Phi}{\partial s} - \frac{1}{\rho} \frac{\partial p}{\partial s} \right\} - (2\boldsymbol{\Omega} \times \mathbf{u}_h) \cdot \mathbf{k}. \quad (\text{B58})$$

20 Similarly, from (B38),

$$\left(\frac{\partial \mathbf{u}_h}{\partial t}\right)_s = -w(\mathbf{u}_h \cdot \nabla \mathbf{k}) + \left[ \left(\frac{\partial z}{\partial t}\right)_s - w \right] \left(\frac{\partial s}{\partial z}\right) \frac{\partial \mathbf{u}_h}{\partial s} - \zeta_h \mathbf{k} \times \mathbf{u}_h - \nabla_z(K_2 + \Phi) - \frac{1}{\rho} \nabla_z p - [2\boldsymbol{\Omega} \times \mathbf{u}]_z. \quad (\text{B59})$$

Observe that both of these equations simplify when  $w = (\partial z / \partial t)_s$ , i.e. model levels are advected with the vertical wind.

An alternative form of these equation can similarly be obtained in terms of  $\dot{s}$ . Substituting (B44) into (B58) then gives

$$\left(\frac{\partial w}{\partial t}\right)_s = \mathbf{u}_h \cdot (\mathbf{u}_h \cdot \nabla \mathbf{k}) - \mathbf{u}_h \cdot \nabla_s w - \dot{s} \frac{\partial w}{\partial s} + \left(\frac{\partial s}{\partial z}\right) \left[ -\frac{\partial \Phi}{\partial s} - \frac{1}{\rho} \frac{\partial p}{\partial s} \right] - (2\boldsymbol{\Omega} \times \mathbf{u}_h) \cdot \mathbf{k}. \quad (\text{B60})$$

Similarly, substituting (B44) into (B59) and using the identity

$$(\mathbf{u}_h \cdot \nabla_s z) \left(\frac{\partial s}{\partial z}\right) \frac{\partial \mathbf{u}_h}{\partial s} = (\nabla_s \times \mathbf{u}_h) \times \mathbf{u}_h - \zeta_h \mathbf{k} \times \mathbf{u}_h + (\nabla_s z) \frac{\partial K_2}{\partial z} \quad (\text{B61})$$

5 then gives

$$\left(\frac{\partial \mathbf{u}_h}{\partial t}\right)_s = -w(\mathbf{u}_h \cdot \nabla \mathbf{k}) - \nabla_s K_2 - \zeta_s \mathbf{k} \times \mathbf{u}_h - \dot{s} \frac{\partial \mathbf{u}_h}{\partial s} - \nabla_z \Phi - \frac{1}{\rho} \nabla_z p - [2\boldsymbol{\Omega} \times \mathbf{u}]_z, \quad (\text{B62})$$

where

$$\nabla_s \times \mathbf{u}_h = \mathbf{k} \zeta_s, \quad \text{and} \quad \zeta_s = \mathbf{k} \cdot (\nabla_s \times \mathbf{u}_h). \quad (\text{B63})$$

In this case the vertical advection terms are removed when  $\dot{s} = 0$ , i.e. the vertical coordinate is advected with the 3D wind  $\mathbf{u}$ .

10 Note that under the shallow-atmosphere approximation, the first metric terms (those that include  $(\mathbf{u}_h \cdot \nabla \mathbf{k})$ ) in (B58)-(B62) are typically dropped.

## B8 3D Vector Invariant Form

From (B22) and (B39) the evolution equation for the 3D velocity vector takes the form

$$\left(\frac{\partial \mathbf{u}}{\partial t}\right)_s = \left(\frac{\partial z}{\partial t}\right)_s (\mathbf{k} \cdot \nabla \mathbf{u}) - \nabla(K + \Phi) - \frac{1}{\rho} \nabla p - (\boldsymbol{\zeta} + 2\boldsymbol{\Omega}) \times \mathbf{u} \quad (\text{B64})$$

15 Then taking the dot product of this expression with  $\mathbf{k}$  gives

$$\left(\frac{\partial w}{\partial t}\right)_s = \left(\frac{\partial s}{\partial z}\right) \left[ \left(\frac{\partial z}{\partial t}\right)_s \frac{\partial w}{\partial s} - \frac{\partial}{\partial s}(K + \Phi) - \frac{1}{\rho} \frac{\partial p}{\partial s} \right] - [(\boldsymbol{\zeta} + 2\boldsymbol{\Omega}) \times \mathbf{u}] \cdot \mathbf{k} \quad (\text{B65})$$

where we have used  $\mathbf{k} \cdot (\mathbf{k} \cdot \nabla \mathbf{u}) = \mathbf{k} \cdot \nabla w$ . Similarly, the prognostic equation for horizontal velocity from (B33) is reformulated as

$$\left(\frac{\partial \mathbf{u}_h}{\partial t}\right)_s = \left(\frac{\partial z}{\partial t}\right)_s \left(\frac{\partial s}{\partial z}\right) \frac{\partial \mathbf{u}_h}{\partial s} - \nabla_z(K + \Phi) - \frac{1}{\rho} \nabla_z p - [(\boldsymbol{\zeta} + 2\boldsymbol{\Omega}) \times \mathbf{u}]_z. \quad (\text{B66})$$

20 Note that the vorticity term in this expression can be simplified further using

$$[(\zeta + 2\Omega) \times \mathbf{u}]_z = -(\zeta_h + \mathbf{k} \cdot 2\Omega)(\mathbf{u}_h \times \mathbf{k}) - w\mathbf{k} \times (\zeta + 2\Omega), \quad (\text{B67})$$

and

$$-\mathbf{k} \times \zeta = \mathbf{k} \cdot \nabla \mathbf{u} - \nabla(\mathbf{k} \cdot \mathbf{u}) + \mathbf{u} \cdot \nabla \mathbf{k} = \frac{\partial \mathbf{u}_h}{\partial z} - \nabla_z w + \mathbf{u}_h \cdot \nabla \mathbf{k}. \quad (\text{B68})$$

## B9 Covariant Component Formulation

In conjunction with (B41), the horizontal momentum equation (in 2D vector invariant form as (B59) or (B62), or in 3D vector invariant form as (B66)) with an arbitrary vertical coordinate gives rise to a two-term pressure gradient. This can be avoided by prognosing the covariant components of the velocity in place of the physical velocity components. We define a horizontal covariance operator by

$$5 \quad [\mathbf{b}]_s \equiv [\mathbf{b}]_z + (\nabla_s z)(\mathbf{k} \cdot \mathbf{b}). \quad (\text{B69})$$

Applying this operator to the horizontal velocity gives

$$\mathbf{v}_h \equiv [\mathbf{u}]_s = \mathbf{u}_h + (\nabla_s z)w. \quad (\text{B70})$$

For a time-dependent  $s$  coordinate, we obtain the identity

$$\left[ \frac{\partial}{\partial t} (\nabla_s z) \right]_s = \nabla_s \left( \frac{\partial z}{\partial t} \right)_s, \quad (\text{B71})$$

10 and so can write

$$\left( \frac{\partial \mathbf{v}_h}{\partial t} \right)_s = \left( \frac{\partial \mathbf{u}_h}{\partial t} \right)_s + (\nabla_s z) \left( \frac{\partial w}{\partial t} \right)_s + w \nabla_s \left( \frac{\partial z}{\partial t} \right)_s. \quad (\text{B72})$$

Then using (B72), (B65) and (B66) and identity

$$\begin{aligned} \left( \frac{\partial z}{\partial t} \right)_s \left( \frac{\partial s}{\partial z} \right) \left[ \frac{\partial \mathbf{u}_h}{\partial s} + (\nabla_s z) \frac{\partial w}{\partial s} \right] + w \nabla_s \left( \frac{\partial z}{\partial t} \right)_s \\ = \left( \frac{\partial s}{\partial z} \right) \left( \frac{\partial z}{\partial t} \right)_s \left\{ \frac{\partial \mathbf{v}_h}{\partial s} - \nabla_s \left[ \left( \frac{\partial s}{\partial z} \right)^{-1} w \right] \right\} + \nabla_s \left[ \left( \frac{\partial z}{\partial t} \right)_s w \right] \end{aligned} \quad (\text{B73})$$

15 gives

$$\left( \frac{\partial \mathbf{v}_h}{\partial t} \right)_s = -\nabla_s \left[ K - w \left( \frac{\partial z}{\partial t} \right)_s + \Phi \right] - \frac{1}{\rho} \nabla_s p - [(\zeta + 2\Omega) \times \mathbf{u}]_s \quad (\text{B74})$$

$$+ \left( \frac{\partial s}{\partial z} \right) \left( \frac{\partial z}{\partial t} \right)_s \left\{ \frac{\partial \mathbf{v}_h}{\partial s} - \nabla_s \left[ \left( \frac{\partial s}{\partial z} \right)^{-1} w \right] \right\}. \quad (\text{B75})$$

Finally, we can expand the vorticity term and hence obtain

$$\begin{aligned} \left( \frac{\partial \mathbf{v}_h}{\partial t} \right)_s = -\nabla_s \left[ K - w \left( \frac{\partial z}{\partial t} \right)_s + \Phi \right] - \frac{1}{\rho} \nabla_s p - [2\Omega \times \mathbf{u}]_s \\ + [\mathbf{k} \cdot \nabla_s \times \mathbf{v}_h] (\mathbf{u}_h \times \mathbf{k}) - \dot{s} \left\{ \frac{\partial \mathbf{v}_h}{\partial s} - \nabla_s \left[ \left( \frac{\partial s}{\partial z} \right)^{-1} w \right] \right\} - \left( \frac{\partial z}{\partial t} \right)_s \mathbf{u}_h \cdot \nabla \mathbf{k}. \end{aligned} \quad (\text{B76})$$

## B10 Vorticity-Divergence Form

The vorticity-divergence form of the dynamical equations in an arbitrary vertical coordinate predicts the absolute vorticity ( $\zeta_h^*$ ) and velocity divergence ( $D$ ) given by

$$\zeta_h^* = (\nabla_s \times \mathbf{u}_h + 2\Omega) \cdot \mathbf{k}, \quad (\text{B77})$$

and

$$D \equiv \nabla_s \cdot \mathbf{u}_h, \quad (\text{B78})$$

5 respectively, instead of the horizontal velocity. The horizontal velocity can be obtained from the streamfunction  $\psi$  and the velocity potential  $\chi$  following

$$\mathbf{u}_h = \mathbf{k} \times \nabla_s \psi + \nabla_s \chi. \quad (\text{B79})$$

By using (B79) in (B77) and (B78), we obtain the elliptic equations that diagnose the streamfunction and velocity potential from the predicted velocity and divergence as

$$10 \quad \nabla_s^2 = \zeta_h^* - 2\Omega \cdot \mathbf{k}, \quad \text{and} \quad \nabla_s^2 \chi = D, \quad (\text{B80})$$

respectively.

By taking the material derivative (B46) of (B77) and using the horizontal momentum equation (B38), (B79) and (B80), the absolute vorticity prediction equation emerges,

$$\left( \frac{\partial \zeta_h^*}{\partial t} \right)_s - J_s(\zeta_h^*, \psi) + \nabla_s \cdot (\zeta_h^* \nabla_s \chi) + \nabla_s \cdot \left( \dot{s} \frac{\partial}{\partial s} \nabla_s \psi \right) + \mathbf{k} \cdot \nabla_s \times \left( \dot{s} \frac{\partial}{\partial s} \nabla_s \chi \right) + J_s(\rho^{-1}, p) = 0, \quad (\text{B81})$$

15 where  $J_s(a, b) = \mathbf{k} \cdot \nabla_s \times (a \nabla_s b)$  is the Jacobian operator. It can also be shown that  $\dot{s}$  relates to the vertical velocity  $w$  through

$$\dot{s} = \left( \frac{\partial s}{\partial z} \right) (w - w_c), \quad (\text{B82})$$

where

$$w_c \equiv \left( \frac{\partial z}{\partial t} \right)_s + (\mathbf{k} \times \nabla_s \psi + \nabla_s \chi) \cdot (\nabla_s z). \quad (\text{B83})$$

By taking the material derivative of (B78) and using (B38), (B79) and (B80), we can obtain the divergence prediction 20 equation

$$\left( \frac{\partial D}{\partial t} \right)_s - J_s(\zeta_h^*, \chi) - \nabla_s \cdot (\zeta_h^* \nabla_s \psi) + \nabla_s \cdot \left( \dot{s} \frac{\partial}{\partial s} \nabla_s \chi \right) + \left( \mathbf{k} \times \frac{\partial}{\partial s} \nabla_s \psi \right) \cdot \nabla_s \dot{s} \quad (\text{B84})$$

$$+ \nabla_s \cdot (\nabla_s K_2 + g \nabla_s z) + \nabla_s \cdot \left( \frac{1}{\rho} \nabla_s p \right) = 0, \quad (\text{B85})$$

where  $K_2$  can be reformulated in terms of streamfunction and velocity potential as

$$K_2 = \frac{1}{2} [\nabla_s \cdot (\psi \nabla_s \psi) - \psi \nabla_s^2 \psi + \nabla_s \cdot (\chi \nabla_s \chi) - \chi \nabla_s^2 \chi] + J_s(\psi, \chi). \quad (\text{B86})$$

## 25 B11 Momentum Form

The momentum form of the prognostic equations emerges by combining the prognostic velocity equations with a continuity equation. Essentially any of the continuity equations can be chosen, as long as the mass field represented by the equation is everywhere non-zero. However, the most common options are moist psuedo-density (Ullrich and Jablonowski, 2012a) or dry pseudo-density (Skamarock et al., 2012). Here we denote our density variable by  $\tilde{\rho}_s$ , and assume no external sources or sinks of  $\tilde{\rho}$ . Multiplying (B60) through by  $\tilde{\rho}_s$  and using (B55) gives

$$\left( \frac{\partial \tilde{\rho}_s w}{\partial t} \right)_s = \tilde{\rho}_s \mathbf{u}_h \cdot (\mathbf{u}_h \cdot \nabla \mathbf{k}) - \nabla_s \cdot (\tilde{\rho}_s \mathbf{u}_h w) - \frac{\partial}{\partial s} (\tilde{\rho}_s \dot{s} w) + \tilde{\rho}_s \left( \frac{\partial s}{\partial z} \right) \left[ -\frac{\partial \Phi}{\partial s} - \frac{1}{\rho} \frac{\partial p}{\partial s} \right] - (2\boldsymbol{\Omega} \times \tilde{\rho}_s \mathbf{u}_h) \cdot \mathbf{k}. \quad (\text{B87})$$

Similarly, from (B62) we have

$$\left( \frac{\partial \tilde{\rho}_s \mathbf{u}_h}{\partial t} \right)_s = -\tilde{\rho}_s w (\mathbf{u}_h \cdot \nabla \mathbf{k}) - \tilde{\rho}_s \nabla_s K_2 - \zeta_s \mathbf{k} \times \tilde{\rho}_s \mathbf{u}_h - \mathbf{u}_h \nabla_s \cdot (\tilde{\rho}_s \mathbf{u}_h) - \frac{\partial}{\partial s} (\tilde{\rho}_s \dot{s} \mathbf{u}_h) - \tilde{\rho}_s \left( \nabla_z \Phi + \frac{1}{\rho} \nabla_z p \right) - [2\boldsymbol{\Omega} \times \tilde{\rho}_s \mathbf{u}]_z, \quad (\text{B88})$$

[Blue text to be moved to paper 2]

## 10 Appendix C: Idealized Physical Parameterizations

### C1 Kessler Physics

The cloud microphysics update according to the following equation set:

$$\frac{\Delta \theta}{\Delta t} = -\frac{L}{c_p \pi} \left( \frac{\Delta q_{vs}}{\Delta t} + E_r \right) \quad (\text{C1})$$

$$\frac{\Delta q_v}{\Delta t} = \frac{\Delta q_{vs}}{\Delta t} + E_r \quad (\text{C2})$$

$$15 \quad \frac{\Delta q_c}{\Delta t} = -\frac{\Delta q_{vs}}{\Delta t} - A_r - C_r \quad (\text{C3})$$

$$\frac{\Delta q_r}{\Delta t} = -E_r + A_r + C_r - V_r \frac{\partial q_r}{\partial z}, \quad (\text{C4})$$

where  $L$  is the latent heat of condensation,  $A_r$  is the autoconversion rate of cloud water to rain water,  $C_r$  is the collection rate of rain water,  $E_r$  is the rain water evaporation rate, and  $V_r$  is the rain water terminal velocity.

The pressure follows from the equation of state

$$20 \quad p = \rho R_d T (1 + 0.61 q_v) \quad (\text{C5})$$

with  $p$  the pressure,  $\rho$  the density of moist air,  $R_d$  the gas constant for dry air,  $T$  the temperature and  $q_v$  the mixing ratio of water vapor. The equation is rewritten as a nondimensional pressure  $\Pi$  equation.

$$\pi = \left( \frac{p}{p_0} \right)^{\frac{R_d T}{c_p}} \quad (\text{C6})$$

To determine the saturation vapor mixing ratio the Teten's formula is used,

$$q_{vs}(p, T) = \left( \frac{380.0}{p} \right) \exp \left( 17.27 \times \frac{T - 273.0}{T - 36.0} \right) \quad (\text{C7})$$

The autoconvection rate ( $A_r$ ) and collection rate ( $C_r$ ) follow Kessler parametrization and are defined by:

$$A_r = k_1(q_c - a) \quad (\text{C8})$$

$$5 \quad C_r = k_2 q_c q_r^{0.875} \quad (\text{C9})$$

With  $k_1 = 0.001\text{s}^{-1}$ ,  $a = 0.001\text{g.g}^{-1}$  and  $k_2 = 2.2\text{s}^{-1}$

Deriving from Klemp and Wilhelmson (1978) description of cloud water,rain water and water vapor mixing ratios. they are define as followed:

$$q_c^{n+1} = \max(q_c^r - \Delta q_r, 0) \quad (\text{C10})$$

10

$$q_r^{n+1} = \max(q_r^r - \Delta q_r + S, 0) \quad (\text{C11})$$

where  $S$  is the sedimentation term and  $\Delta q_r$  is defined as

$$\Delta q_r = q_c^n - \frac{q_c^n - \Delta t \max(A_r, 0)}{1 + \Delta t C_r} \quad (\text{C12})$$

The Rain evaporation equation is defined similarly to Ogura and Takahashi (1971) description:

$$15 \quad E_r = \frac{1}{\rho} \frac{\left(1 - \frac{q_v}{q_{vs}}\right) C(\rho q_r)^{0.525}}{5.4 \times 10^5 + \frac{2.55 \times 10^6}{pq_{vs}}} \quad (\text{C13})$$

With ventilation factor C define as

$$C_r = 1.6 + 124.9(\rho q_r)^{0.2046} \quad (\text{C14})$$

The liquid water terminal velocity is similar to Soong and Ogura (1973) description with a mean density adjustment as suggested by Kessler (1969):

$$20 \quad V_r = 36349(\rho q_r)^{0.1346} \left( \frac{\rho}{\rho_0} \right)^{-\frac{1}{2}} \quad (\text{C15})$$

## Appendix D: Surface Fluxes on an Aqua-Planet with Prescribed Sea Surface Temperatures

The forcing by surface fluxes from an idealized ocean is described in Reed and Jablonowski (2012) and is partly reproduced here. We use a model configuration which corresponds to an aqua-planet setup with prescribed sea surface temperatures (SSTs). This forcing by the surface fluxes is applied to the state variables in the lowermost model level using a partially implicit

formulation to avoid numerical instabilities. Throughout this section we use the subscript  $a$  to denote variables defined on the lowermost model level.

The surface fluxes depend on the *drag coefficient*  $C_d$ , defined as

$$\begin{aligned} C_d &= C_{d0} + C_{d1}|\mathbf{v}_a| && \text{for } |\mathbf{v}_a| < 20 \text{ m s}^{-1} \\ C_d &= 0.002 && \text{for } |\mathbf{v}_a| \geq 20 \text{ m s}^{-1}, \end{aligned} \quad (\text{D1})$$

where  $C_{d0}$  and  $C_{d1}$  are  $7.0 \times 10^{-4}$  (unitless) and  $6.5 \times 10^{-5} \text{ s m}^{-1}$ , respectively, and  $|\mathbf{v}_a|$  is the magnitude of the horizontal wind at the lowermost model level. In terms of the zonal wind  $u_a$  and meridional wind  $v_a$ , it is defined as

$$|\mathbf{v}_a| = \sqrt{u_a^2 + v_a^2}. \quad (\text{D2})$$

For both evaporation and sensible heat the bulk coefficient is set to

$$C_E = C_H = 0.0011. \quad (\text{D3})$$

The formulation of the surface fluxes makes use of the height of the lowermost full model level  $z_a$  (in m). For pressure-based models,  $z_a$  can be expressed with the help of the hydrostatic equation in terms of pressure

$$z_a = \frac{R_d T_{\nu,a}}{g} \frac{(\ln p_s - \ln p_-)}{2}, \quad (\text{D4})$$

where  $T_{\nu,a} = T_a(1 + 0.608q_a)$  is the virtual temperature at the lowermost full model level and  $p_-$  is the edge pressure at the model level interface between the lowest and second lowest full model levels. This notation and all following equations assume that the temperature, horizontal wind components and the specific humidity in the physical parameterization package are co-located in both the vertical and horizontal directions, as is the case for the Lorenz grid. The height of the lowest full model level should ideally lie between 60-70m above the ground to make the results comparable to those in the literature.

As described in Reed and Jablonowski (2012), the surface fluxes can be written as

$$\frac{\partial \mathbf{v}_a}{\partial t} = -\frac{C_d |\mathbf{v}_a| \mathbf{v}_a}{z_a} \quad (\text{D5})$$

$$\frac{\partial T_a}{\partial t} = \frac{C_H |\mathbf{v}_a| (T_s - T_a)}{z_a} \quad (\text{D6})$$

$$\frac{\partial q_a}{\partial t} = \frac{C_E |\mathbf{v}_a| (q_{sat,s} - q_a)}{z_a}. \quad (\text{D7})$$

We note that the wind at the surface is taken to be zero and therefore does not appear explicitly in (D5). In these equations  $T_s$  denotes the prescribed sea surface temperature (SST) and  $q_{sat,s}$  is the saturation specific humidity defined by the Clausius-Clapeyron equation

$$q_{sat}(p) \approx \varepsilon \frac{e_s(T_s)}{p} \approx \frac{\varepsilon}{p} e_0^* e^{-(L/R_\nu)[(1/T_s) - (1/T_0)]}, \quad (\text{D8})$$

where  $e_0^*$  ( $= 610.78 \text{ Pa}$ ) is the saturation vapor pressure at  $T_0 = 273.16 \text{ K}$ .

The final form of the surface fluxes will vary for models with other choices of prognostic variables. For example, if potential temperature  $\Theta_a$  is used (D6) takes the form

$$\frac{\partial \Theta_a}{\partial t} = \frac{C_H |\mathbf{v}_a| (T_s - T_a)}{z_a} \left( \frac{p_0}{p_a} \right)^{R_a/c_p} \quad (\text{D9})$$

where  $p_0 = 1000$  hPa is a reference pressure. This conversion uses the assumption that the pressure is time-invariant when individual physics parameterizations are applied. For other choices of prognostic variables like  $(\rho u)_a$ ,  $(\rho v)_a$ ,  $(\rho \Theta)_a$  and  $(\rho q)_a$  the right-hand-side of (D5), (D9) and (D7) would need to be multiplied by the density of the air  $\rho$ .

In order to ensure numerical stability, each of the aforementioned surface fluxes are applied via a semi-implicit operator. We demonstrate this procedure on the temperature evolution equation (D6). First, the time derivative is expanded using a backward Euler operator,

$$\frac{T_a^{n+1} - T_a^n}{\Delta t} = \frac{C_H |\mathbf{v}_a^n| (T_s - T_a^{n+1})}{z_a}. \quad (\text{D10})$$

The superscripts  $n$  and  $n + 1$  represent the current time step (after the update from the large-scale condensation scheme) and the future time step, respectively. Note, that on the right-hand-side of the equation the only variable taken implicitly is  $T_a$ .  $|\mathbf{v}_a^n|$  is evaluated at the current time step and  $C_H$  is constant. The equation can now be solved for  $T_a^{n+1}$

$$T_a^{n+1} = \frac{T_a^n + C_H |\mathbf{v}_a^n| T_s \frac{\Delta t}{z_a}}{1 + C_H |\mathbf{v}_a^n| \frac{\Delta t}{z_a}}. \quad (\text{D11})$$

Similar equations for  $\mathbf{v}_a$  and  $q_a$  can be calculated

$$\mathbf{v}_a^{n+1} = \frac{\mathbf{v}_a^n}{1 + C_d^n |\mathbf{v}_a^n| \frac{\Delta t}{z_a}} \quad (\text{D12})$$

$$q_a^{n+1} = \frac{q_a^n + C_E |\mathbf{v}_a^n| q_{sat,s}^n \frac{\Delta t}{z_a}}{1 + C_E |\mathbf{v}_a^n| \frac{\Delta t}{z_a}}, \quad (\text{D13})$$

with the time-level dependent coefficient  $C_d^n$ . Notice that the second term in the numerator of (D12) is absent in the case of the zonal and meridional wind. This is because the wind is set to zero at the surface.

## D1 Simplified Mixing in the Planetary Boundary Layer

The forcing by the planetary boundary layer is described in Reed and Jablonowski (2012) and is partly reproduced here. To parameterize the surface fluxes that impact the zonal velocity  $u$ , the meridional velocity  $v$  and moisture  $q$  we start with the time rate of change equations

$$\frac{\partial u}{\partial t} = -\frac{1}{\rho} \frac{\partial \rho \overline{w' u'}}{\partial z} \quad (\text{D14})$$

$$\frac{\partial v}{\partial t} = -\frac{1}{\rho} \frac{\partial \rho \overline{w' v'}}{\partial z} \quad (\text{D15})$$

$$\frac{\partial q}{\partial t} = -\frac{1}{\rho} \frac{\partial \rho \overline{w' q'}}{\partial z}. \quad (\text{D16})$$

Potential temperature, as opposed to temperature, is used in the boundary layer parameterization because the vertical profile of the potential temperature is a suitable indicator of static stability. This adds the time rate of change equation

$$\frac{\partial \Theta}{\partial t} = -\frac{1}{\rho} \frac{\partial \rho \overline{w' \Theta'}}{\partial z}. \quad (\text{D17})$$

Here  $u'$ ,  $v'$ ,  $w'$ ,  $\Theta'$  and  $q'$  symbolize the deviations of the zonal velocity, meridional velocity, vertical velocity, potential temperature and specific humidity from their averages, respectively. The average is indicated by an overbar. Note, assuming pressure

5 is held constant (which is a common assumption in physical parameterizations), the potential temperature time tendency can be converted back to a temperature tendency of the following form

$$\frac{\partial T}{\partial t} = -\frac{1}{\rho} \left( \frac{p}{p_0} \right)^\kappa \frac{\partial \rho \overline{w' \Theta'}}{\partial z}. \quad (\text{D18})$$

with the reference pressure  $p_0 = 1000$  hPa.

The turbulent mixing is characterized by a constant vertical eddy diffusivity to represent Ekman-like profiles of boundary  
10 layers

$$\overline{w' u'} = -K_m \frac{\partial u}{\partial z} \quad (\text{D19})$$

$$\overline{w' v'} = -K_m \frac{\partial v}{\partial z} \quad (\text{D20})$$

$$\overline{w' \Theta'} = -K_E \frac{\partial \Theta}{\partial z} \quad (\text{D21})$$

$$\overline{w' q'} = -K_E \frac{\partial q}{\partial z}. \quad (\text{D22})$$

15 Here,  $K_m$  is the eddy diffusivity coefficient for momentum and  $K_E$  is the eddy diffusivity coefficient for energy and set equal to that for water vapor. In order to calculate the eddy diffusivity coefficients, the eddy diffusivity is matched to that for the surface flux calculated in Appendix D at the lowermost model level. To allow for a smooth transition above the boundary layer ( $p_{top} = 850$  hPa) the diffusivity coefficients for momentum taper to zero as

$$K_m = C_d |\mathbf{v}_a| z_a \quad \text{for } p > p_{top}$$

$$K_m = C_d |\mathbf{v}_a| z_a \exp \left( - \left[ \frac{p_{top} - p}{p_{strato}} \right]^2 \right) \quad \text{for } p \leq p_{top}. \quad (\text{D23})$$

20 Here the constant  $p_{strato}$  determines the rate of decrease and is set to 100 hPa.  $K_E$  is defined by

$$K_E = C_E |\mathbf{v}_a| z_a \quad \text{for } p > p_{top}$$

$$K_E = C_E |\mathbf{v}_a| z_a \exp \left( - \left[ \frac{p_{top} - p}{p_{strato}} \right]^2 \right) \quad \text{for } p \leq p_{top}. \quad (\text{D24})$$

We suggest implementing the boundary layer scheme with an implicit temporal discretization to avoid numerical instabilities. The details of this discretization are somewhat complicated, and so we refer to implementation details in Appendix D of Reed and Jablonowski (2012). In addition, we supply the DCMIP modeling groups with the complete “simple-physics” package as  
25 used in the model CAM which can serve as a template routine.

## **Appendix E: Conclusions**

[To be written]

*Author contributions.* TEXT

*Acknowledgements.* DCMIP2016 is sponsored by the National Center for Atmospheric Research Computational Information Systems Laboratory, the Department of Energy Office of Science (award no. DE-SC0016015), the National Science Foundation (award no. 1629819), the National Aeronautics and Space Administration (award no. ??), the National Oceanic and Atmospheric Administration (award no. ??), the Office of Naval Research and CU Boulder Research Computing. This work was made possible with support from our student and postdoctoral participants: Sabina Abba Omar, Scott Bachman, Amanda Back, Tobias Bauer, Vinicius Capistrano, Spencer Clark, Ross Dixon, Christopher Eldred, Robert Fajber, Jared Ferguson, Emily Foshee, Ariane Frassoni, Alexander Goldstein, Jorge Guerra, Chasity Henson, Adam Herrington, Tsung-Lin Hsieh, Dave Lee, Theodore Letcher, Weiwei Li, Laura Mazzaro, Maximo Menchaca, Jonathan Meyer, Farshid Nazari, John O'Brien, Bjarke Tobias Olsen, Hossein Parishani, Charles Pelletier, Thomas Rackow, Kabir Rasouli, Cameron Rencurrel, Koichi Sakaguchi, Gökhan Sever, James Shaw, Konrad Simon, Abhishek Srivastava, Nicholas Szapiro, Kazushi Takemura, Pushp Raj Tiwari, Chii-Yun Tsai, Richard Urata, Karin van der Wiel, Lei Wang, Eric Wolf, Zheng Wu, Haiyang Yu, Sungduk Yu and Jiawei Zhuang. We would also like to thank Rich Loft, Cecilia Banner, Kathryn Peczkowicz and Rory Kelly (NCAR), Carmen Ho, Perla Dinger, and Gina Skyberg (UC Davis) and Kristi Hansen (University of Michigan) for administrative support during the workshop and summer school.

## References

- Arakawa, A. and Konor, C. S.: Unification of the anelastic and quasi-hydrostatic systems of equations, *Monthly Weather Review*, 137, 5 710–726, 2009.
- Arakawa, A. and Lamb, V. R.: Computational design of the basic dynamical processes of the UCLA general circulation model, *Methods in computational physics*, 17, 173–265, 1977.
- Arakawa, A. and Moorthi, S.: Baroclinic instability in vertically discrete systems, *Journal of the atmospheric sciences*, 45, 1688–1708, 1988.
- Ascher, U. M., Ruuth, S. J., and Spiteri, R. J.: Implicit-explicit Runge-Kutta methods for time-dependent partial differential equations, 10 *Applied Numerical Mathematics*, 25, 151–167, 1997.
- Bermejo, R. and Conde, J.: A conservative quasi-monotone semi-Lagrangian scheme, *Mon. Wea. Rev.*, 130, 423–430, doi:10.1175/1520-0493(2002)130<0423:ACQMSL>2.0.CO;2, 2002.
- Bermejo, R. and Staniforth, A.: The conversion of semi-Lagrangian advection schemes to quasi-monotone schemes, *Mon. Wea. Rev.*, 120, 2622–2631, doi:10.1175/1520-0493(1992)120<2622:TCOSLA>2.0.CO;2, 1992.
- Charney, J. G. and Phillips, N.: Numerical integration of the quasi-geostrophic equations for barotropic and simple baroclinic flows, *Journal of Meteorology*, 10, 71–99, 1953.
- Colella, P. and Woodward, P. R.: The Piecewise Parabolic Method (PPM) for Gas-Dynamical Simulations, *J. Comput. Phys.*, 54, 174–201, 1984.
- Cotter, C. J. and Shipton, J.: Mixed finite elements for numerical weather prediction, *Journal of Computational Physics*, 231, 7076–7091, 20 2012.
- Davies, T., Cullen, M., Malcolm, A., Mawson, M., Staniforth, A., White, A., and Wood, N.: A new dynamical core for the Met Office’s global and regional modelling of the atmosphere, *Quarterly Journal of the Royal Meteorological Society*, 131, 1759–1782, 2005.
- de Grandpré, J., Tanguay, M., Qaddouri, A., Zerroukat, M., and McLinden, C.: Semi-Lagrangian Advection of Stratospheric Ozone on a Yin-Yang Grid System, *Mon. Wea. Rev.*, 144, 1035–1050, doi:10.1175/MWR-D-15-0142.1, 2016.
- Dennis, J. M., Edwards, J., Evans, K. J., Guba, O., Lauritzen, P. H., Mirin, A. A., St-Cyr, A., Taylor, M. A., and Worley, P. H.: CAM-SE: A scalable spectral element dynamical core for the Community Atmosphere Model, *The International Journal of High Performance Computing Applications*, 26, 74–89, 2012.
- Dipankar, A., Stevens, B., Heinze, R., Moseley, C., Zängl, G., Giorgetta, M., and Brdar, S.: Large eddy simulation using the general circulation model ICON, *Journal of Advances in Modeling Earth Systems*, 7, 963–986, 2015.
- Du, Q., Gunzburger, M. D., and Ju, L.: Constrained centroidal Voronoi tessellations for surfaces, *SIAM Journal on Scientific Computing*, 24, 30 1488–1506, 2003.
- Dubey, S., Dubos, T., Hourdin, F., and Upadhyaya, H. C.: On the inter-comparison of two tracer transport schemes on icosahedral grids, *Applied Mathematical Modelling*, 2015.
- Dubos, T. and Tort, M.: Equations of Atmospheric Motion in Non-Eulerian Vertical Coordinates: Vector-Invariant Form and Quasi- 35 Hamiltonian Formulation, *Monthly Weather Review*, 142, 3860–3880, 2014.
- Dubos, T., Dubey, S., Tort, M., Mittal, R., Meurdesoif, Y., and Hourdin, F.: DYNAMICO-1.0, an icosahedral hydrostatic dynamical core designed for consistency and versatility, *Geoscientific Model Development*, 8, 3131–3150, 2015.
- Gal-Chen, T. and Somerville, R.: Numerical solution of the Navier Stokes equations with topography., *Journal of Computational Physics*, 17, 276–310, doi:10.1016/0021-9991(75)90054-6, 1975.

- Gassmann, A. and Herzog, H. J.: Towards a consistent numerical compressible non-hydrostatic model using generalized Hamiltonian tools, *Quarterly Journal of the Royal Meteorological Society*, 134, 1597–1613, doi:10.1002/qj.297, 2008.
- 5 Giraldo, F. X., Kelly, J. F., and Constantinescu, E.: Implicit-explicit formulations of a three-dimensional nonhydrostatic unified model of the atmosphere (NUMA), *SIAM Journal on Scientific Computing*, 35, B1162–B1194, 2013.
- Girard, C., Plante, A., Desgagné, M., McTaggart-Cowan, R., Côté, J., Charron, M., Gravel, S., Lee, V., Patoine, A., Qaddouri, A., Roch, M., Spacek, L., Tanguay, M., Vaillancourt, P., and Zadra, A.: Staggered Vertical Discretization of the Canadian Environmental Multiscale (GEM) model using a coordinate of the log-hydrostatic-pressure type, *Mon. Wea. Rev.*, 142, 1183–1196, 2014.
- 10 Guba, O., Taylor, M., and St-Cyr, A.: Optimization-based limiters for the spectral element method, *Journal of Computational Physics*, 267, 176–195, 2014.
- Guerra, J. E. and Ullrich, P. A.: A High-order Staggered Finite-Element Vertical Discretization for Non-Hydrostatic Atmospheric Models, *Geoscientific Model Development*, doi:10.5194/gmd-2015-275, (in press), 2016.
- Harris, L. M. and Lin, S.-J.: A Two-Way Nested Global-Regional Dynamical Core on the Cubed-Sphere Grid, *Monthly Weather Review*, 141, 283–306, 2013.
- 15 Harris, L. M., Lin, S.-J., and Tu, C.: High resolution climate simulations using GFDL HiRAM with a stretched global grid, *Journal of Climate*, 2016.
- Holdaway, D., Thuburn, J., and Wood, N.: Comparison of Lorenz and Charney–Phillips vertical discretisations for dynamics–boundary layer coupling. Part I: Steady states, *Quarterly Journal of the Royal Meteorological Society*, 139, 1073–1086, 2013a.
- 20 Holdaway, D., Thuburn, J., and Wood, N.: Comparison of Lorenz and Charney–Phillips vertical discretisations for dynamics–boundary layer coupling. Part II: Transients, *Quarterly Journal of the Royal Meteorological Society*, 139, 1087–1098, 2013b.
- Hortal, M. and Simmons, A.: Use of reduced Gaussian grids in spectral models, *Mon. Weather Rev.*, 119, 1057–1074, 1991.
- Jablonowski, C. and Williamson, D. L.: The pros and cons of diffusion, filters and fixers in atmospheric general circulation models, in: *Numerical Techniques for Global Atmospheric Models*, pp. 381–493, Springer, 2011.
- 25 Jablonowski, C., Lauritzen, P. H., Taylor, M. A., and Nair, R. D.: Idealized test cases for the dynamical cores of atmospheric general circulation models: a proposal for the NCAR ASP 2008 summer colloquium, *Tech. rep.*, 2008.
- Kageyama, A. and Sato, T.: The Yin-Yang grid: An overset grid in spherical geometry, *Geochem. Geophys. Geosyst.*, 5, doi:10.1029/2004GC000734, 2004.
- Kasahara, A.: Various vertical coordinate systems used for numerical weather prediction, *Monthly Weather Review*, 102, 509–522, 1974.
- 30 Kessler, E.: On the distribution and continuity of water substance in atmospheric circulation, *Meteorological Monographs*, 32, 82–84, 1969.
- Klemp, J. B.: A terrain-following coordinate with smoothed coordinate surfaces, *Monthly weather review*, 139, 2163–2169, 2011.
- Klemp, J. B. and Wilhelmson, R. B.: The simulation of three-dimensional convective storm dynamics, *J. Atmos. Sci.*, 35, 1070–1096, 1978.
- Klemp, J. B., Skamarock, W. C., and Dudhia, J.: Conservative split-explicit time integration methods for the compressible nonhydrostatic equations, *Monthly Weather Review*, 135, 2897–2913, 2007.
- 35 Klemp, J. B., Dudhia, J., and Hassiotis, A. D.: An upper gravity-wave absorbing layer for NWP applications, *Monthly Weather Review*, 136, 3987–4004, doi:10.1175/2008MWR2596.1, 2008.
- Kühnlein, C. and Smolarkiewicz, P. K.: An unstructured-mesh finite-volume MPDATA for compressible atmospheric dynamics, *J. Comput. Phys.*, 334, 16–30, 2017.
- Kühnlein, C., Smolarkiewicz, P. K., and Dörnbrack, A.: Modelling atmospheric flows with adaptive moving meshes, *J. Comput. Phys.*, 231, 2741–2763, 2012.

- Kurowski, M. J., Grabowski, W. W., and Smolarkiewicz, P. K.: Anelastic and compressible simulation of moist deep convection, *Journal of the Atmospheric Sciences*, 71, 3767–3787, 2014.
- 5 Laprise, R.: The Euler equations of motion with hydrostatic pressure as an independent variable, *Monthly weather review*, 120, 197–207, 1992.
- Lin, S.: A finite-volume integration method for computing pressure gradient force in general vertical coordinates, *Q. J. Roy. Meteor. Soc.*, 123, 1749–1762, 1997.
- Lin, S.-J.: A “vertically Lagrangian” finite-volume dynamical core for global models, *Monthly Weather Review*, 132, 2293–2307, 2004.
- 10 Lin, S.-J. and Rood, R.: An explicit flux-form semi-Lagrangian shallow-water model on the sphere., *Q.J.R.Meteorol.Soc.*, 123, 2477–2498, 1997.
- Lloyd, S.: Least squares quantization in PCM, *IEEE transactions on information theory*, 28, 129–137, 1982.
- Lock, S.-J., Bitzer, H.-W., Coals, A., Gadian, A., and Mobbs, S.: Demonstration of a cut-cell representation of 3D orography for studies of atmospheric flows over very steep hills, *Monthly Weather Review*, 140, 411–424, 2012.
- 15 Lorenz, E. N.: Energy and numerical weather prediction, *Tellus*, 12, 364–373, 1960.
- Malardel, S., Wedi, N., Deconinck, W., Diamantakis, M., Kühnlein, C., Mozdynski, G., Hamrud, M., and Smolarkiewicz, P.: A new grid for the IFS, *ECMWF Newsletter*, 146, 23–28, 2016.
- Miura, H.: An upwind-biased conservative advection scheme for spherical hexagonal–pentagonal grids, *Monthly Weather Review*, 135, 4038–4044, 2007.
- 20 Niwa, Y., Tomita, H., Satoh, M., and Imasu, R.: A three-dimensional icosahedral grid advection scheme preserving monotonicity and consistency with continuity for atmospheric tracer transport, *Journal of the Meteorological Society of Japan. Ser. II*, 89, 255–268, 2011.
- Ogura, Y. and Takahashi, T.: Numerical simulation of the life cycle of a thunderstorm cell, *Mon. Weather Rev.*, 99, 895–911, 1971.
- Okabe, A., Boots, B., Sugihara, K., and Chiu, S. N.: Spatial tessellations: concepts and applications of Voronoi diagrams, vol. 501, John Wiley & Sons, 2009.
- 25 Phillips, N. A.: The equations of motion for a shallow rotating atmosphere and the “traditional approximation”, *Journal of the Atmospheric Sciences*, 23, 626–628, 1966.
- Prusa, J. M. and Smolarkiewicz, P. K.: An all-scale anelastic model for geophysical flows: dynamic grid deformation, *J. Comput. Phys.*, 190, 601–622, 2003.
- Prusa, J. M., Smolarkiewicz, P. K., and Wyszogrodzki, A. A.: EULAG, a computational model for multiscale flows, *Computers & Fluids*, 37, 1193–1207, 2008.
- Putman, W. M. and Lin, S.-J.: Finite-volume transport on various cubed-sphere grids, *J. Comput. Phys.*, 227, 55–78, 2007.
- 30 Qaddouri, A. and Lee, V.: The Canadian Global Environmental Multiscale model on the Yin-Yang grid system, *Q. J. R. Meteorol. Soc.*, 137, 1913–1926, 2011.
- Qaddouri, A., Laayouni, L., Loisel, S., Côté, J., and Gander, M.: Optimized Schwarz methods with an overset grid for the shallow-water equations: Preliminary results, *Appl. Numer. Math.*, 58, 459–471, 2008.
- Randall, D. A.: Geostrophic adjustment and the finite-difference shallow-water equations, *Monthly Weather Review*, 122, 1371–1377, 1994.
- Reed, K. A. and Jablonowski, C.: Idealized tropical cyclone simulations of intermediate complexity: a test case for AGCMs, *J. Adv. Model. Earth Syst.*, 4, M04 001, doi:10.1029/2011MS000099, 2012.
- Ringler, T., Ju, L., and Gunzburger, M.: A multiresolution method for climate system modeling: application of spherical centroidal Voronoi tessellations, *Ocean Dynamics*, 58, 475–498, 2008.

- Ringler, T. D. and Randall, D. A.: The ZM grid: an alternative to the Z grid, *Monthly weather review*, 130, 1411–1422, 2002.
- Ringler, T. D., Thuburn, J., Klemp, J. B., and Skamarock, W. C.: A unified approach to energy conservation and potential vorticity dynamics  
5 for arbitrarily-structured C-grids, *Journal of Computational Physics*, 229, 3065–3090, 2010.
- Ronchi, C., Ianoco, R., and Paolucci, P. S.: The 'Cubed Sphere': a new method for the solution of partial differential equations in spherical geometry, *J. Comput. Phys.*, 124, 93–114, doi:10.1006/jcph.1996.0047, 1996.
- Sadourny, R.: Conservative finite-difference approximations of the primitive equation on quasi-uniform spherical grids, *Mon. Weather Rev.*, 100, 136–144, doi:10.1175/1520-0493(1972)100<0136:CFAOTP>2.3.CO;2, 1972.
- 10 Satoh, M.: Conservative scheme for the compressible nonhydrostatic models with the horizontally explicit and vertically implicit time integration scheme, *Monthly weather review*, 130, 1227–1245, 2002.
- Satoh, M.: Conservative scheme for a compressible nonhydrostatic model with moist processes, *Monthly weather review*, 131, 1033–1050, 2003.
- 15 Schär, C., Leuenberger, D., Fuhrer, O., Lüthi, D., and Girard, C.: A new terrain-following vertical coordinate formulation for atmospheric prediction models, *Monthly Weather Review*, 130, 2459–2480, 2002.
- Schmidt, F.: Variable fine mesh in spectral global models, *Beitr. Phys. Atmos.*, 50, 211–217, 1977.
- Simmons, A. J. and Burridge, D. M.: An energy and angular-momentum conserving vertical finite-difference scheme and hybrid vertical coordinates, *Monthly Weather Review*, 109, 758–766, 1981.
- Skamarock, W. C. and Gassmann, A.: Conservative transport schemes for spherical geodesic grids: High-order flux operators for ODE-based  
20 time integration, *Monthly Weather Review*, 139, 2962–2975, 2011.
- Skamarock, W. C. and Klemp, J. B.: The stability of time-split numerical methods for the hydrostatic and the nonhydrostatic elastic equations, *Monthly Weather Review*, 120, 2109–2127, 1992.
- Skamarock, W. C., Klemp, J. B., Duda, M. G., Fowler, L. D., Park, S.-H., and Ringler, T. D.: A multiscale nonhydrostatic atmospheric model using centroidal Voronoi tesselations and C-grid staggering, *Monthly Weather Review*, 140, 3090–3105, 2012.
- 25 Smolarkiewicz, P. and Szmelter, J.: A nonhydrostatic unstructured-mesh soundproof model for simulation of internal gravity waves, *Acta Geophys.*, 59, 1109–1134, 2011.
- Smolarkiewicz, P. K. and Szmelter, J.: MPDATA: an edge-based unstructured-grid formulation, *J. Comput. Phys.*, 206, 624–649, 2005.
- Smolarkiewicz, P. K., Temperton, C., Thomas, S. J., and Wyszogrodzki, A. A.: Spectral Preconditioners for nonhydrostatic atmospheric models: extreme applications, In: Proceedings of the ECMWF seminar series on recent developments in numerical methods for atmospheric  
30 and ocean modelling. Reading, UK, pp. 203–220, 2004.
- Smolarkiewicz, P. K., Kühnlein, C., and Wedi, N. P.: A consistent framework for discrete integrations of soundproof and compressible PDEs of atmospheric dynamics, *J. Comput. Phys.*, 263, 185–205, 2014.
- Smolarkiewicz, P. K., Deconinck, W., Hamrud, M., Kühnlein, C., Mozdzynski, G., Szmelter, J., and Wedi, N. P.: A finite-volume module for simulating global all-scale atmospheric flows, *J. Comput. Phys.*, 314, 287–304, 2016.
- 35 Smolarkiewicz, P. K., Kühnlein, C., and Grabowski, W.: A finite-volume module for cloud-resolving simulations of global atmospheric flows, *J. Comput. Phys.*, in review, 2017a.
- Smolarkiewicz, P. K., Kühnlein, C., and Grabowski, W.: A finite-volume module for cloud-resolving simulations of global atmospheric flows, submitted to *J. Comput. Phys.*, 2017b.
- Soong, S.-T. and Ogura, Y.: A comparison between axisymmetric and slab-symmetric cumulus cloud models, *J. Atmos. Sci.*, 30, 879–893, 1973.

- Sørenson, B., Kaas, E., and Korsholm, U. S.: A mass-conserving and multi-tracer efficient transport scheme in the online integrated Enviro-HIRLAM model, *Geosci. Model Dev.*, 6, 1029–1042, doi:10.5194/gmd-6-1029-2013, 2013.
- 5 Staniforth, A. and Thuburn, J.: Horizontal grids for global weather and climate prediction models: a review, *Quarterly Journal of the Royal Meteorological Society*, 138, 1–26, 2012.
- Taylor, M. A. and Fournier, A.: A compatible and conservative spectral element method on unstructured grids, *Journal of Computational Physics*, 229, 5879–5895, 2010.
- Thuburn, J.: A PV-based shallow-water model on a hexagonal–icosahedral grid, *Monthly Weather Review*, 125, 2328–2347, 1997.
- 10 Thuburn, J. and Woollings, T.: Vertical discretizations for compressible Euler equation atmospheric models giving optimal representation of normal modes, *Journal of Computational Physics*, 203, 386–404, 2005.
- Thuburn, J., Ringler, T. D., Skamarock, W. C., and Klemp, J. B.: Numerical representation of geostrophic modes on arbitrarily structured C-grids, *Journal of Computational Physics*, 228, 8321–8335, 2009.
- Tokioka, T.: Some considerations on vertical differencing, *Meteorological Society of Japan Journal*, 56, 98–111, 1978.
- 15 Tomita, H. and Satoh, M.: A new dynamical framework of nonhydrostatic global model using the icosahedral grid, *Fluid Dynamics Research*, 34, 357–400, 2004.
- Tomita, H., Tsugawa, M., Satoh, M., and Goto, K.: Shallow water model on a modified icosahedral geodesic grid by using spring dynamics, *Journal of Computational Physics*, 174, 579–613, 2001.
- Tomita, H., Satoh, M., and Goto, K.: An optimization of the icosahedral grid modified by spring dynamics, *Journal of Computational Physics*, 20 183, 307–331, 2002.
- Ullrich, P.: Understanding the treatment of waves in atmospheric models. Part 1: The shortest resolved waves of the 1D linearized shallow-water equations, *Quarterly Journal of the Royal Meteorological Society*, 140, 1426–1440, 2014a.
- Ullrich, P. A.: A global finite-element shallow-water model supporting continuous and discontinuous elements, *Geosci. Model Dev.*, 7, 3017–3035, 2014b.
- 25 Ullrich, P. A. and Jablonowski, C.: MCORE: A non-hydrostatic atmospheric dynamical core utilizing high-order finite-volume methods, *Journal of Computational Physics*, 231, 5078–5108, 2012a.
- Ullrich, P. A. and Jablonowski, C.: Operator-split Runge–Kutta–Rosenbrock methods for nonhydrostatic atmospheric models, *Monthly Weather Review*, 140, 1257–1284, 2012b.
- Ullrich, P. A., Jablonowski, C., Kent, J., Lauritzen, P. H., Nair, R. D., and Taylor, M. A.: Dynamical core model intercomparison project 30 (DCMIP) test case document, Tech. rep., 2012.
- Weller, H., Lock, S.-J., and Wood, N.: Runge–Kutta IMEX schemes for the horizontally explicit/vertically implicit (HEVI) solution of wave equations, *Journal of Computational Physics*, 252, 365–381, 2013.
- Wicker, L. J. and Skamarock, W. C.: Time-splitting methods for elastic models using forward time schemes, *Monthly weather review*, 130, 2088–2097, 2002.
- 35 Zängl, G.: The ICON (ICOsaHedral Non-hydrostatic) modelling framework of DWD and MPI-M: Description of the non-hydrostatic dynamical core, *Quarterly Journal of the Royal Meteorological Society*, 141, 563–579, doi:10.1002/qj.2378, 2014.
- Zängl, G., Reinert, D., Rípodas, P., and Baldauf, M.: The ICON (ICOsaHedral Non-hydrostatic) modelling framework of DWD and MPI-M: Description of the non-hydrostatic dynamical core, *Quarterly Journal of the Royal Meteorological Society*, 141, 563–579, 2015.