

DCMIP2016, Part 1: Models and Equation Sets

Paul A. Ullrich¹, Christiane Jablonowski², James Kent³, Peter H. Lauritzen⁴, Ramachandran Nair⁴, Kevin A. Reed⁵, Colin M. Zarzycki⁴, David A. Hall⁶, Alex Reinecke⁷, Kevin Viner⁷, Don Dazlich⁸, Ross Heikes⁸, Celal Konor⁸, David Randall⁸, Xi Chen⁹, Lucas Harris⁹, Marco Giorgetta¹⁰, Daniel Reinert¹¹, Christian Kuehnlein¹², Robert Walko¹³, Vivian Lee¹⁴, Abdessamad Qaddouri¹⁴, Claude Girard¹⁴, Hiroaki Miura¹⁵, Tomoki Ohno¹⁵, Ryuji Yoshida¹⁶, Joseph Klemp⁴, Sang-Hun Park⁴, William Skamarock⁴, Thomas Dubos¹⁷, Yann Meurdesoif¹⁷, Elijah Goodfriend¹⁸, and Hans Johansen¹⁸

¹University of California, Davis

²University of Michigan

³University of South Wales

⁴National Center for Atmospheric Research

⁵Stony Brook University

⁶University of Colorado, Boulder

⁷Naval Research Laboratory

⁸Colorado State University

⁹Geophysical Fluid Dynamics Laboratory

¹⁰Max Planck Institute for Meteorology

¹¹Deutscher Wetterdienst (DWD)

¹²European Center for Medium-Range Weather Forecasting

¹³University of Miami

¹⁴Environment Canada

¹⁵University of Tokyo

¹⁶RIKEN

¹⁷Institut Pierre-Simon Laplace (IPSL)

¹⁸Lawrence Berkeley National Laboratory

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

Abstract. This paper provides a comprehensive review of the design of modern non-hydrostatic atmospheric dynamical cores, including relevant equation sets, numerical stabilization techniques and idealized physics routines.

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 Navier-Stokes equations) throughout the atmosphere, 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, vertical coordinates, representation of topography, numerical method, physics/dynamics coupler, 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 has 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 previous intercomparison

5 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 is 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.

10 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 already begun to leverage this information to improve their own models, which will in turn positively impact the quality of weather and climate simulations going forward.

15 The proposed workshop has advanced our knowledge of (a) the relative behaviors exhibited by atmospheric dynamical cores, (b) differences that arise among mechanisms for coupling the physical parameterizations and dynamical core, and (c) 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
20 set of simulations that can be used for assessment of any future dynamical core.

2 Notation

2.1 List of Symbols

Table 1 lists the symbols used in this paper.

2.2 List of Physical Constants

25 A list of physical constants which are used throughout this document is given in Table 2. Constants which are specific to each test case are similarly tabulated at the beginning of each section.

2.3 Great Circle Distance

The great circle distance is used throughout the document and is given by

$$R_c(\lambda_1, \varphi_1; \lambda_2, \varphi_2) = a \arccos(\sin \varphi_1 \sin \varphi_2 + \cos \varphi_1 \cos \varphi_2 \cos(\lambda_1 - \lambda_2)). \quad (1)$$

Table 1. List of symbols used in this manuscript

Symbol	Description
λ	Longitude (in radians)
φ	Latitude (in radians)
z	Height with respect to mean sea level (set to zero)
p_s	Surface pressure (p_s of moist air if $q > 0$)
Φ_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
ω	Vertical pressure velocity
δ	Divergence
ζ	Relative vorticity
p	Pressure (pressure of moist air if $q > 0$)
ρ	Total air density
ρ_d	Dry air density
T	Temperature
T_v	Virtual temperature
Θ	Potential temperature
Θ_v	Virtual potential temperature
q	Specific humidity
P_{ls}	Large-scale precipitation rate
q_c	Cloud water mixing ratio
q_r	Rain water mixing ratio

3 Model Grids

Model grids are a primary concern for dynamical cores, since they determine how degrees of freedom are distributed across the globe. 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. This section describes many of the model grids that have been employed globally.
5

Table 2. A list of physical constants used in this document.

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

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 [Figure]. 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, 2004b). 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) 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 (2014). On the equiangular cubed-sphere grid, coordinates are given as (α, β, 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 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 20 equilateral spherical triangles, 30 great circle edges and 12 vertices. These initial triangles are then subdivided repeatedly

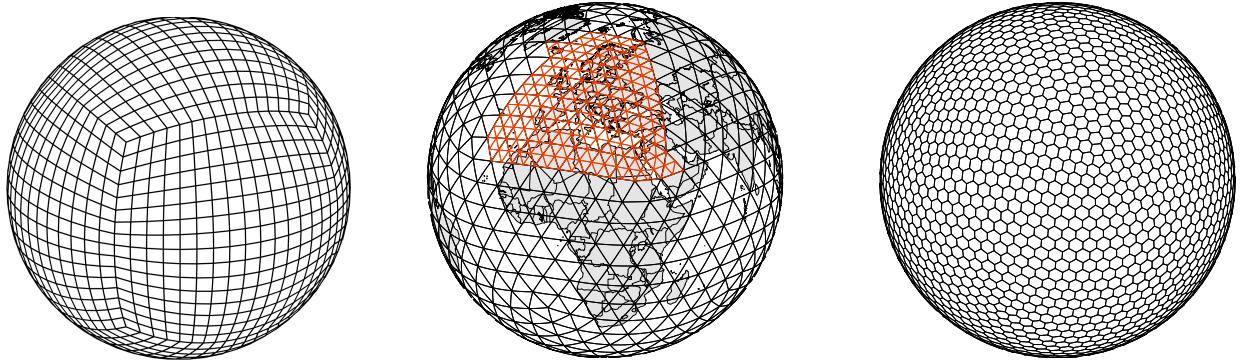


Figure 1. (Left) A cubed-sphere grid with 16 elements along each cubed-sphere edge. (Center) An icosahedral (triangular) grid with additional refinement over Europe, as indicated in red. (Right) An icosahedral (hexagonal) grid.

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 brakes the initial isotropy of the icosahedron and results in non-equilateral triangles of different sizes.

5 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 ?. An example of such a
10 grid is depicted in Figure 1 (center). For a discussion of the effective resolution see ?. 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 combination with a refined region over Europe at 6.5 km
15 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
20 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, fa-

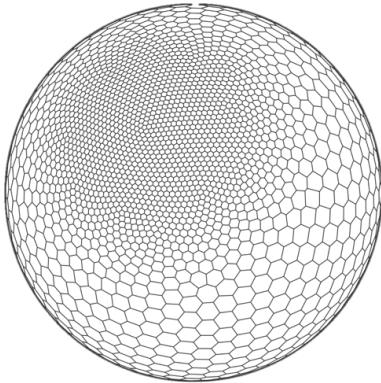


Figure 2. A constrained centroidal Voronoi tessellation grid with localized grid density that could be employed in the MPAS model.

cilitating 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 node-neighbors (cells that share a given node).

- 5 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.

3.5 Constrained Centroidal Voronoi Tessellation (CCVT) grids

- 10 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 Ω_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, for an arbitrary set of generators it is easy to produce highly distorted polygons, particularly if the density of generators
15 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). 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.
20 However, CCVT grids are fundamentally constructed using a very different technique and will often contain polygonal elements with more than six sides.

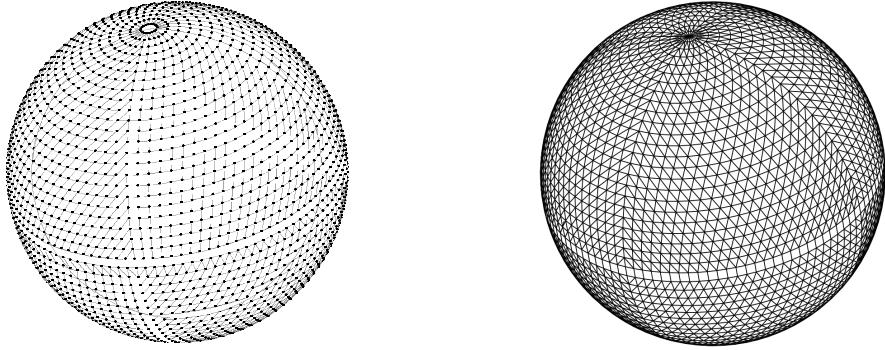


Figure 3. Locations of the octahedral reduced Gaussian grid nodes (left), and the edges of the primary mesh connecting the nodes as applied with the finite-volume discretisation in FVM (right). A coarse grid with only 24 latitudes between pole and equator is used for illustration. The dual mesh resolution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

3.6 Octahedral reduced Gaussian grid

As with the classical reduced Gaussian grid of ?, the octahedral reduced Gaussian grid (??) 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 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 7.11).

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

$$(-\frac{\pi}{4} - \delta_\theta \leq \theta \leq \frac{\pi}{4} + \delta_\theta) \cap (-\frac{3\pi}{4} - \delta_\lambda \leq \lambda \leq \frac{3\pi}{4} + \delta_\lambda), \quad (2)$$

15 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 $\delta_\theta = 2$ degrees and $\delta_\lambda = 3\delta_\theta$. The Yang component covers an analogous area, but is rotated 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.

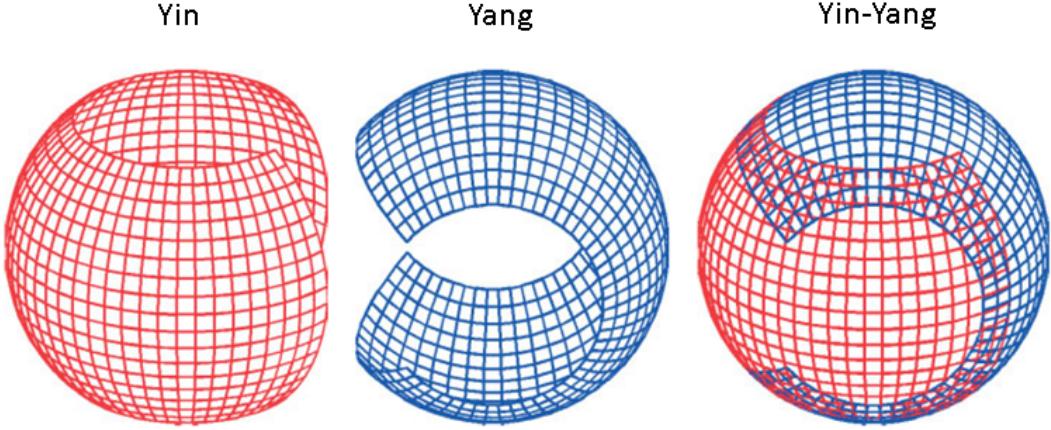


Figure 4. Yin-Yang grid

4 Moist Nonhydrostatic Equation Sets

In this section we describe 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

$$\frac{d}{dt} = \frac{\partial}{\partial t} + \mathbf{u} \cdot \nabla. \quad (3)$$

- 5 Note that tracer variables q_i , including specific humidity q , satisfy the simple Lagrangian relationship

$$\frac{dq_i}{dt} = 0. \quad (4)$$

4.1 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

$$10 \quad R^* = R_d + (R_w - R_d)q, \quad c_p^* = c_{pd} + (c_{pw} - c_{pd})q, \quad c_v^* = c_{vd} + (c_{vw} - c_{vd})q. \quad (5)$$

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 ρ_d , moist density ρ , pressure p , vapor pressure e , temperature T , virtual temperature T_v ,
15 Exner pressure π , potential temperature θ , and virtual potential temperature θ_v . Common ratios $\kappa = R^*/c_p^*$, $\epsilon = R_d/R_w$, and $\gamma = c_p^*/c_v^*$ 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

$$p = \rho R_d T_v, \quad \pi = \left(\frac{p}{p_0} \right)^\kappa, \quad \theta = T \left(\frac{p_0}{p} \right)^\kappa, \quad \theta_v = T_v \left(\frac{p_0}{p} \right)^\kappa, \quad (6)$$

which further give rise to

$$5 \quad p = \left(\frac{\rho R_d \theta_v}{p_0^\kappa} \right)^\gamma, \quad \pi = \left(\frac{\rho R_d \theta_v}{p_0} \right)^{R^*/c_p^*}, \quad \theta = \frac{T}{\pi}, \quad \theta_v = \frac{T_v}{\pi}. \quad (7)$$

Note that virtual temperature is typically approximated by

$$T_v \approx T \left(1 + \frac{(1-\epsilon)}{\epsilon} q \right), \quad (8)$$

which arises from the exact relationship

$$T_v = \frac{T}{1 - \frac{\epsilon}{p}(1-\epsilon)}, \quad (9)$$

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

4.2 Prognostic equations for thermodynamic variables

Note that, as a consequence of (4), the following simplifications can be applied:

$$\frac{1}{T_v} \frac{dT_v}{dt} = \frac{1}{T} \frac{dT}{dt}, \quad \frac{dR^*}{dt} = 0, \quad \frac{dc_p^*}{dt} = 0, \quad \frac{dc_v^*}{dt} = 0. \quad (10)$$

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

$$15 \quad \frac{d\rho}{dt} = -\rho \nabla \cdot \mathbf{u}, \quad (11)$$

or equivalently in the Eulerian frame,

$$\frac{\partial \rho}{\partial t} = -\nabla \cdot (\rho \mathbf{u}). \quad (12)$$

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

$$\frac{1}{T} \frac{dT}{dt} - \frac{\kappa}{p} \frac{dp}{dt} = \frac{J}{c_p^*}, \quad (13)$$

20 which can be alternatively written as

$$\frac{d\theta}{dt} = \frac{J\theta}{c_p^*}, \quad \text{or} \quad \frac{d\theta_v}{dt} = \frac{J\theta_v}{c_p^*}. \quad (14)$$

These equations can then be combined with (11) to obtain

$$\frac{\partial}{\partial t} (\rho \theta_v) + \nabla \cdot (\rho \theta_v \mathbf{u}) = \frac{J \rho \theta_v}{c_p^*}, \quad (15)$$

or similarly for θ . In conjunction with the material derivative of the ideal gas law,

$$\frac{1}{p} \frac{dp}{dt} = \frac{1}{\rho} \frac{d\rho}{dt} + \frac{1}{T_v} \frac{dT_v}{dt}, \quad (16)$$

the thermodynamic equation can be written in the form

$$\frac{c_v^*}{R^* T_v} \frac{dT_v}{dt} - \frac{1}{\rho} \frac{d\rho}{dt} = \frac{J}{R^*}. \quad (17)$$

5 Then substituting (11) gives a prognostic equation for virtual temperature,

$$\frac{c_v^*}{R^*} \frac{dT_v}{dt} + T_v \nabla \cdot \mathbf{u} = \frac{JT_v}{R^*}. \quad (18)$$

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,

$$\frac{c_v^*}{c_p^*} \frac{dp}{dt} + p \nabla \cdot \mathbf{u} = \frac{Jp}{c_p^*}. \quad (19)$$

10 And similarly for Exner pressure,

$$\frac{c_v^*}{R^*} \frac{d\pi}{dt} + \pi \nabla \cdot \mathbf{u} = \frac{J\pi}{c_p^*}. \quad (20)$$

4.3 Momentum Equations

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

$$\frac{d\mathbf{u}}{dt} = \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\frac{1}{\rho} \nabla p - 2\boldsymbol{\Omega} \times \mathbf{u} - \nabla \Phi, \quad (21)$$

15 where $\boldsymbol{\Omega}$ denotes the planetary vorticity vector and Φ 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 [citation needed]. Note that it is common to rewrite the pressure gradient force using the relationship

$$20 \quad -\frac{1}{\rho} \nabla p = -\frac{R_d c_p^*}{R^*} \theta_v \nabla \pi, \quad (22)$$

which follows from (6). A second form of (21) emerges on substituting the vector calculus identity

$$\mathbf{u} \cdot \nabla \mathbf{u} = \nabla K + \boldsymbol{\zeta} \times \mathbf{u}, \quad (23)$$

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

$$25 \quad \frac{\partial \mathbf{u}}{\partial t} = -\frac{1}{\rho} \nabla p - \nabla(K + \Phi) - (\boldsymbol{\zeta} + 2\boldsymbol{\Omega}) \times \mathbf{u}. \quad (24)$$

Because no gradients of vectors appear in this equation, it avoids derivatives of the coordinate basis that would emerge in (21).

In conjunction with (11), both (21) also give rise to the flux-form momentum equations,

$$\frac{\partial}{\partial t}(\rho \mathbf{u}) = -\nabla \cdot (\mathbf{u} \otimes \mathbf{u} + \mathcal{I}p) - 2\Omega \times (\rho \mathbf{u}) - \rho \nabla \Phi, \quad (25)$$

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

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

$$\Phi = ga^2 \left[\frac{1}{a} - \frac{1}{a+z} \right], \quad \text{and} \quad \Omega = \Omega(\mathbf{k} \sin \phi + \mathbf{j} \cos \phi), \quad (26)$$

where g is gravitational acceleration at the surface, a is the radius of the planet, Ω is the rotation rate (in s^{-1}), ϕ 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
10 don't utilize a height-based vertical coordinate, the geopotential is generally treated as a prognostic variable, with an evolution equation that emerges from the definition $w = dz/dt$,

$$\frac{d\Phi}{dt} = \frac{a^2 gw}{(a+z)^2}. \quad (27)$$

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

$$\Phi = gz, \quad \text{and} \quad \Omega = \Omega \sin \phi \mathbf{k}, \quad (28)$$

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

$$\frac{d\Phi}{dt} = gw. \quad (29)$$

4.4 Orthogonal Formulation

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

$$20 \quad [\mathbf{b}]_z = \mathbf{b} - (\mathbf{b} \cdot \mathbf{k})\mathbf{k}. \quad (30)$$

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

$$\mathbf{u} = \mathbf{u}_h + w\mathbf{k}, \quad (31)$$

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

$$25 \quad \frac{d}{dt} = \frac{\partial}{\partial t} + \mathbf{u}_h \cdot \nabla + w\mathbf{k} \cdot \nabla. \quad (32)$$

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

$$\frac{d}{dt} = \frac{\partial}{\partial t} + \mathbf{u}_h \cdot \nabla_z + w \frac{\partial}{\partial z}. \quad (33)$$

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 (24) $\cdot \mathbf{k}$ expands as

$$5 \quad \frac{\partial w}{\partial t} = -\frac{1}{\rho} \frac{\partial p}{\partial z} - \frac{\partial}{\partial z} (K + \Phi) - [(\zeta + 2\Omega) \times \mathbf{u}] \cdot \mathbf{k}, \quad (34)$$

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

$$\frac{\partial \mathbf{u}_h}{\partial t} = -\frac{1}{\rho} \nabla_z p - \nabla_z (K + \Phi) - [(\zeta + 2\Omega) \times \mathbf{u}]_z. \quad (35)$$

It is also 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

$$10 \quad \mathbf{u} \cdot \nabla \mathbf{u} = \mathbf{u}_h \cdot \nabla \mathbf{u}_h + (\nabla \times \mathbf{u}_h) \times (w\mathbf{k}) + (\mathbf{u} \cdot \nabla w)\mathbf{k}. \quad (36)$$

The first term in this expression admits the relationships

$$[\mathbf{u}_h \cdot \nabla \mathbf{u}_h]_z = \nabla_z K_2 + \zeta_h \mathbf{k} \times \mathbf{u}_h, \quad (37)$$

$$(\mathbf{u}_h \cdot \nabla \mathbf{u}_h) \cdot \mathbf{k} = -\mathbf{u}_h \cdot (\mathbf{u}_h \cdot \nabla \mathbf{k}) = -K_2 (\nabla \cdot \mathbf{k}) - \frac{1}{2} [\mathbf{u}_h \cdot (\nabla \times \mathbf{u}_t) + (\nabla \times \mathbf{u}_h) \cdot \mathbf{u}_t] \quad (38)$$

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

15 metric terms associated with horizontal advection of \mathbf{k} which must be accounted for.

Derivation.

$$\mathbf{u} \cdot \nabla \mathbf{u} = \mathbf{u}_h \cdot \nabla \mathbf{u}_h + \mathbf{u}_h \cdot \nabla(w\mathbf{k}) + w\mathbf{k} \cdot \nabla \mathbf{u}_h + w\mathbf{k} \cdot \nabla(w\mathbf{k}) \quad (39)$$

$$= \mathbf{u}_h \cdot \nabla \mathbf{u}_h + \nabla(\mathbf{u}_h \cdot (w\mathbf{k})) - (w\mathbf{k}) \times (\nabla \times \mathbf{u}_h) - \mathbf{u}_h \times (\nabla \times w\mathbf{k}) + \nabla(\frac{1}{2}w^2) + ((\nabla \times (w\mathbf{k})) \times (w\mathbf{k})) \quad (40)$$

$$= \mathbf{u}_h \cdot \nabla \mathbf{u}_h + (\nabla \times \mathbf{u}_h) \times (w\mathbf{k}) + (\mathbf{u} \cdot \nabla w)\mathbf{k} \quad (41)$$

20 where we have used

$$(\nabla \times (w\mathbf{k})) \times (w\mathbf{k}) = (w(\nabla \times \mathbf{k}) + (\nabla w) \times \mathbf{k}) \times (w\mathbf{k}) = -w\mathbf{k} \times (\nabla w \times \mathbf{k}) = (-w\mathbf{k} \cdot \mathbf{k}) \nabla w - (-w\mathbf{k} \cdot \nabla w)\mathbf{k} \quad (42)$$

$$-\mathbf{u}_h \times (\nabla \times w\mathbf{k}) = -\mathbf{u}_h \times (w(\nabla \times \mathbf{k}) + (\nabla w) \times \mathbf{k}) = -\mathbf{u}_h \times (\nabla w \times \mathbf{k}) = (\mathbf{u}_h \cdot \nabla w)\mathbf{k} \quad (43)$$

Thus the vertical velocity equation, obtained by taking (21) $\cdot \mathbf{k}$, leading to

$$25 \quad \frac{\partial w}{\partial t} = \mathbf{u}_h \cdot (\mathbf{u}_h \cdot \nabla \mathbf{k}) - \mathbf{u} \cdot \nabla w - \frac{1}{\rho} \frac{\partial p}{\partial z} - \frac{\partial \Phi}{\partial z} - (2\Omega \times \mathbf{u}) \cdot \mathbf{k} \quad (44)$$

and

$$\frac{\partial \mathbf{u}_h}{\partial t} = -\frac{1}{\rho} \nabla_z p - \nabla_z (K_2 + \Phi) - (\nabla \times \mathbf{u}_h) \times (w\mathbf{k}) - \zeta_h \mathbf{k} \times \mathbf{u}_h - [2\Omega \times \mathbf{u}]_z. \quad (45)$$

Note that under the shallow atmosphere approximation, the metric term $\mathbf{u}_h \cdot (\mathbf{u}_h \cdot \nabla \mathbf{k})$ in (44) is set equal to zero in accordance with Phillips (1966).

4.5 Arbitrary vertical coordinates

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 ? (hereafter K74). Since \mathbf{x} and t are shared between the two coordinate systems, the chain rule can be applied to obtain expressions

$$5 \quad \frac{\partial}{\partial z} = \frac{\partial s}{\partial z} \frac{\partial}{\partial s}, \quad \nabla_s = \nabla_z + (\nabla_s z)(\mathbf{k} \cdot \nabla), \quad \left(\frac{\partial}{\partial t} \right)_s = \frac{\partial}{\partial t} + \left(\frac{\partial z}{\partial t} \right)_s (\mathbf{k} \cdot \nabla), \quad (46)$$

which correspond to derivatives in the vertical, in the horizontal and in time. This final expression is used to describe the rate of change of a quantity on s surfaces. When applied to s , these operators yields the useful identities

$$\nabla_z s = - \left(\frac{\partial s}{\partial z} \right) \nabla_s z, \quad \text{and} \quad \frac{\partial s}{\partial t} = - \frac{\partial s}{\partial z} \left(\frac{\partial z}{\partial t} \right)_s. \quad (47)$$

From here (46) also gives rise to

$$10 \quad \nabla_z = \nabla_s - \frac{\partial s}{\partial z} (\nabla_s z) \frac{\partial}{\partial s}. \quad (48)$$

This equation can be used directly to rewrite (35) or (45) in terms of derivatives over s . Note that the operators ∇_z and ∇_s are usually introduced in the context of 2D flows, however the construction described here has the advantage of working seamlessly in a 3D context, while admitting the properties $\mathbf{k} \cdot \nabla_z = 0$ and $\mathbf{k} \cdot \nabla_s = 0$.

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

$$15 \quad \dot{s} \equiv \frac{ds}{dt} = \frac{\partial s}{\partial t} + \mathbf{u} \cdot \nabla s = \mathbf{u}_h \cdot \nabla_z s + \left[w - \left(\frac{\partial z}{\partial t} \right)_s \right] \frac{\partial s}{\partial z}. \quad (49)$$

Then using (46) and (49) to rewrite (33) gives an expression for the material derivative for scalars on s surfaces,

$$\frac{dA}{dt} = \left(\frac{\partial A}{\partial t} \right)_s - \frac{\partial s}{\partial z} \left(\frac{\partial z}{\partial t} \right)_s \frac{\partial A}{\partial s} + \mathbf{u}_h \cdot \left[\nabla_s A + (\nabla_z s) \frac{\partial A}{\partial s} \right] + w \frac{\partial s}{\partial z} \frac{\partial A}{\partial s} \quad (50)$$

$$= \left(\frac{\partial A}{\partial t} \right)_s + \mathbf{u}_h \cdot \nabla_s A + \dot{s} \frac{\partial A}{\partial s} \quad (51)$$

The prognostic equation for vertical velocity from (34) is then reformulated using (46),

$$20 \quad \left(\frac{\partial w}{\partial t} \right)_s = \frac{\partial s}{\partial z} \left[\left(\frac{\partial z}{\partial t} \right)_s \frac{\partial w}{\partial s} - \frac{1}{\rho} \frac{\partial p}{\partial s} - \frac{\partial}{\partial s} (K + \Phi) \right] - [(\zeta + 2\Omega) \times \mathbf{u}_h] \cdot \mathbf{k}. \quad (52)$$

Similarly, the prognostic equation for horizontal velocity from (35) is reformulated as

$$\left(\frac{\partial \mathbf{u}_h}{\partial t} \right)_s = - \frac{1}{\rho} \nabla_z p - \nabla_z (K + \Phi) - [(\zeta + 2\Omega) \times \mathbf{u}]_z + \left(\frac{\partial z}{\partial t} \right)_s (\mathbf{k} \cdot \nabla \mathbf{u}_h). \quad (53)$$

Note that the last term of this expression can then be expanded using

$$(\mathbf{k} \cdot \nabla \mathbf{u}_h) = - \mathbf{u}_h \cdot \nabla \mathbf{k} - \mathbf{k} \times (\nabla \times \mathbf{u}_h), \quad (54)$$

25 where under the shallow-atmosphere approximation ($\mathbf{u}_h \cdot \nabla \mathbf{k}$) is typically dropped.

The prognostic equations utilizing horizontal kinetic energy K_2 in place of K are similarly derived from (44),

$$\left(\frac{\partial w}{\partial t}\right)_s = \mathbf{u}_h \cdot (\mathbf{u}_h \cdot \nabla \mathbf{k}) - \mathbf{u}_h \cdot \nabla w + \left(\frac{\partial s}{\partial z}\right) \left\{ \left[\left(\frac{\partial z}{\partial t}\right)_s - w \right] \frac{\partial w}{\partial s} - \frac{1}{\rho} \frac{\partial p}{\partial s} - \frac{\partial}{\partial s} (K_2 + \Phi) \right\} - (2\boldsymbol{\Omega} \times \mathbf{u}_h) \cdot \mathbf{k}, \quad (55)$$

and from (45),

$$\left(\frac{\partial \mathbf{u}_h}{\partial t}\right)_s = -\frac{1}{\rho} \nabla_z p - \nabla_z (K_2 + \Phi) - \left[\left(\frac{\partial z}{\partial t}\right)_s - w \right] \mathbf{k} \times (\nabla \times \mathbf{u}_h) - \zeta_h \mathbf{k} \times \mathbf{u}_h - [2\boldsymbol{\Omega} \times \mathbf{u}]_z - w(\mathbf{u}_h \cdot \nabla \mathbf{k}). \quad (56)$$

- 5 Observe that both of these equations are simplified if $(\partial z/\partial t)_s = w$.

[Is there an easy way to write down the curl (specifically relative vorticity) in arbitrary vertical coordinates?]

4.6 Covariant Formulation

In conjunction with (48), the horizontal momentum equation (either (35) or (45)) 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
10 the physical velocity components. The horizontal partial covariance operator is defined by

$$[\mathbf{b}]_s \equiv [\mathbf{b}]_z + (\nabla_s z)(\mathbf{k} \cdot \mathbf{b}). \quad (57)$$

Hence the covariant horizontal velocity is given by

$$\mathbf{v}_h \equiv [\mathbf{u}]_s = \mathbf{u}_h + (\nabla_s z)w. \quad (58)$$

For a time-dependent s coordinate,

$$15 \quad \left[\frac{\partial}{\partial t} (\nabla_s z) \right]_s = \nabla_s \left(\frac{\partial z}{\partial t} \right)_s, \quad (59)$$

and so, using (52) and (56) the evolution equation for the covariant horizontal velocity reads

$$\left(\frac{\partial \mathbf{v}_h}{\partial t}\right)_s = \left(\frac{\partial \mathbf{u}_h}{\partial t}\right)_s + \left(\frac{\partial w}{\partial t}\right)_s (\nabla_s z) + w \nabla_s \left(\frac{\partial z}{\partial t}\right)_s \quad (60)$$

$$= -\frac{1}{\rho} \nabla_z p - \nabla_z (K_2 + \Phi) - \left[\left(\frac{\partial z}{\partial t}\right)_s - w \right] \mathbf{k} \times (\nabla \times \mathbf{u}_h) - \zeta_h \mathbf{k} \times \mathbf{u}_h - [2\boldsymbol{\Omega} \times \mathbf{u}]_z - w(\mathbf{u}_h \cdot \nabla \mathbf{k}) \quad (61)$$

$$+ (\nabla_s z) \left\{ \frac{\partial s}{\partial z} \left[\left(\frac{\partial z}{\partial t}\right)_s \frac{\partial w}{\partial s} - \frac{1}{\rho} \frac{\partial p}{\partial s} - \frac{\partial}{\partial s} (K_2 + \Phi) - w \frac{\partial w}{\partial s} \right] - [(\zeta + 2\boldsymbol{\Omega}) \times \mathbf{u}] \cdot \mathbf{k} \right\} \quad (62)$$

$$20 \quad + w \nabla_s \left(\frac{\partial z}{\partial t} \right)_s \quad (63)$$

$$= -\frac{1}{\rho} \nabla_s p - \nabla_s (K_2 + \Phi) + w \nabla_s \left(\frac{\partial z}{\partial t} \right)_s - w(\mathbf{u}_h \cdot \nabla \mathbf{k}) - [(\zeta + 2\boldsymbol{\Omega}) \times \mathbf{u}]_s \quad (64)$$

$$- \left[\left(\frac{\partial z}{\partial t}\right)_s - w \right] \left[\mathbf{k} \times (\nabla \times \mathbf{u}_h) - (\nabla_s z) \left(\frac{\partial s}{\partial z} \right) \frac{\partial w}{\partial s} \right].$$

Again, observe that this equation simplifies significantly if $(\partial z/\partial t)_s = w$.

4.7 Conservation Laws in Arbitrary Vertical Coordinates

In this section we rewrite the continuity equation in arbitrary vertical coordinates. The divergence of the contravariant velocity field on surfaces of constant s can be obtained from (46), which is valid for both shallow- and deep-atmosphere formulations and equal to K74 eq. (3.17),

$$5 \quad \nabla_s \cdot \mathbf{u}_h = \nabla_z \cdot \mathbf{u}_h + \left(\frac{\partial s}{\partial z} \right) (\nabla_s z) \cdot \left(\frac{\partial \mathbf{u}_h}{\partial s} \right). \quad (65)$$

On the sphere the 3D divergence takes the form

$$\nabla \cdot \mathbf{u} = \nabla_z \cdot \mathbf{u}_h + \frac{1}{\alpha} \frac{\partial}{\partial z} (\alpha w), \quad (66)$$

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

$$10 \quad \frac{1}{\alpha} \frac{\partial}{\partial z} (\alpha w) = \frac{\partial w}{\partial z} + \frac{w}{\alpha} \frac{\partial \alpha}{\partial z} = \frac{\partial w}{\partial z} + \frac{1}{\alpha} \frac{d\alpha}{dt}. \quad (67)$$

Using (47) to rewrite (49) gives rise to

$$w = \left(\frac{\partial z}{\partial t} \right)_s + \mathbf{u}_h \cdot \nabla_s z + \dot{s} \left(\frac{\partial s}{\partial z} \right)^{-1}, \quad (68)$$

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

$$\frac{\partial w}{\partial z} = \left(\frac{\partial s}{\partial z} \right) \left[\frac{d}{dt} \left(\frac{\partial s}{\partial z} \right)^{-1} + \left(\frac{\partial \mathbf{u}_h}{\partial s} \right) \cdot (\nabla_s z) \right] + \frac{\partial \dot{s}}{\partial s} = 0. \quad (69)$$

15 Substituting this expression into the continuity equation (11), and using (66), (67), and (69) then leads to

$$\frac{d}{dt} \left[\alpha \left(\frac{\partial s}{\partial z} \right)^{-1} \rho \right] + \alpha \left(\frac{\partial s}{\partial z} \right)^{-1} \rho \left[\nabla_z \cdot \mathbf{u}_h + \left(\frac{\partial s}{\partial z} \right) \left(\frac{\partial \mathbf{u}_h}{\partial s} \right) \cdot (\nabla_s z) \right] + \alpha \left(\frac{\partial s}{\partial z} \right)^{-1} \rho \frac{\partial \dot{s}}{\partial s} = 0. \quad (70)$$

Defining the *pseudodensity* as

$$\rho_s = \alpha \left(\frac{\partial s}{\partial z} \right)^{-1} \rho, \quad (71)$$

and using (51) in the form

$$20 \quad \frac{d\rho_s}{dt} = \left(\frac{d\rho_s}{dt} \right)_s + \mathbf{u}_h \cdot \nabla_s \rho_s + \dot{s} \frac{\partial \rho_s}{\partial s}, \quad (72)$$

along with (65) leads to

$$\left(\frac{d\rho_s}{dt} \right)_s + \nabla_s \cdot (\rho_s \mathbf{u}_h) + \frac{\partial}{\partial s} (\rho_s \dot{s}) = 0. \quad (73)$$

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

$$\frac{\partial \rho_s q}{\partial t} + \nabla_s \cdot (\rho_s q \mathbf{u}_h) + \frac{\partial}{\partial s} (\rho_s q \dot{s}) = 0. \quad (74)$$

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

$$\frac{\partial \rho_s \theta_v}{\partial t} + \nabla_s \cdot (\rho_s \theta_v \mathbf{u}_h) + \frac{\partial}{\partial s} (\rho_s \theta_v \dot{s}) = \frac{J \rho_s \theta_v}{c_p^*}. \quad (75)$$

4.8 Examples of vertical coordinates

4.8.1 Terrain-following z coordinates

In general, terrain-following coordinates are adopted so that the topographic surface is also a coordinate surface. This simplifies the treatment of the bottom boundary condition. Common terrain-following coordinates include Gal-Chen and Somerville 5 (1975), which takes the form

$$s = z_{\text{top}} \left(\frac{z - z_s}{z_{\text{top}} - z_s} \right). \quad (76)$$

Another coordinate with smoother variation away from the surface was also introduced by Schär et al. (2002).

4.8.2 Mass-based coordinates

Mass-based coordinates are a generalization of pressure-based coordinates to non-hydrostatic models, with a vertical coordinate 10 defined as the total gravity-weighted overhead mass,

$$s = \int_z^{\infty} \rho g dz. \quad (77)$$

Under this definition,

$$\frac{\partial s}{\partial z} = -\rho g. \quad (78)$$

4.8.3 GEM zeta coordinate

15 [Fix notation]

The vertical coordinate of GEM model (denoted ζ in the model documentation) is of a log-hydrostatic-pressure type and is defined by

$$\log \pi_h = s + B(s)\sigma_p, \quad B(s) = \left(\frac{s - s_T}{(\log p_0) - s_T} \right)^r, \quad \sigma_p = \log \left(\frac{\pi_h}{p_0} \right), \quad (79)$$

with $s_T = \log \pi_T$ and where r is a variable exponent providing added freedom for adjusting the thickness of model layers over 20 high terrain. Figure 5 shows the Charney-Phillips grid in GEM model, giving the position occupied by each variable in the vertical domain. Horizontal velocity \mathbf{v}_h , geopotential Φ and $q = \log(p/\pi_h)$ are on momentum levels. Temperature T , vertical velocity w , generalized vertical velocity \dot{s} and tracers are on thermodynamic levels.

4.9 Vertical staggering

Define Unstaggered, Lorenz and Charney-Phillips staggering here with figure

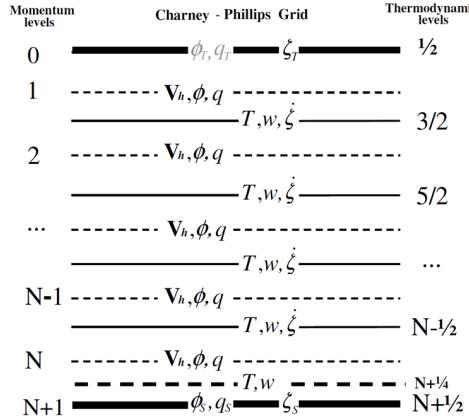


Figure 5. A modification of the Charney-Phillips grid found in GEM

4.10 Model Implementations

4.10.1 Tempest

Tempest is a shallow-atmosphere Eulerian model with terrain-following z -coordinates with prognostic density (72), virtual potential temperature (75), and vector-invariant form for horizontal momentum with $(\partial z / \partial t)_s = 0$ (??) and vertical momentum (34). 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 $c_v^* \approx c_{vd}$.

4.10.2 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 \tilde{s} , mass-weighted tracers (potential temperature, water species), geopotential Φ , “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 Φ and in vector-invariant form for “horizontal” momentum.

4.10.3 ICON

ICON is a shallow-atmosphere Eulerian model based on Gassmann and Herzog (2008) with terrain-following z -coordinates with prognostic density (72), virtual potential temperature (75), and vector-invariant form for horizontal momentum in vector-invariant form (35) and vertical momentum (34). As with tempest, specific heat capacities and ideal gas constant are approximated by their dry values. The model also uses a prognostic equation for Exner pressure to simplify vertical sound wave

propagation, given by

$$\frac{\partial \pi}{\partial t} + \frac{R_d}{c_{vd}} \frac{\pi}{\rho \theta_v} \nabla \cdot (\mathbf{u}_h \rho \theta_v) = \hat{Q}, \quad (80)$$

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 .

4.10.4 FV³

The hydrostatic FV³ model uses a mass-based floating Lagrangian coordinate along with the shallow-atmosphere approximation (Lin, 2004a). Prognostic equations include horizontal momentum (??) in 2D vector-invariant form (45), pseudo-density (72), and virtual potential temperature (75). The non-hydrostatic model further incorporates prognostic geopotential (29) and vertical momentum (44).

4.10.5 GEM

GEM is a semi-Lagrangian model with Eulerian log-hydrostatic-pressure coordinate π_h described in section 4.8.3 (Girard et al., 2014). The non-hydrostatic equations are written explicitly as deviations from hydrostatic balance, using

$$\mu = \frac{\partial p}{\partial \pi_h} - 1, \quad (81)$$

in which case

$$\frac{d\mathbf{v}_h}{dt} + f\mathbf{k} \times \mathbf{v}_h + R_d T_v \nabla_\zeta \log p + (1 + \mu) \nabla_\zeta \Phi = 0, \quad (82)$$

$$\frac{dw}{dt} - g\mu = 0, \quad (83)$$

$$\frac{d}{dt} \ln \left(\pi \frac{\partial \ln \pi}{\partial \zeta} \right) + \nabla_\zeta \cdot \mathbf{V}_h + \frac{\partial \dot{\zeta}}{\partial \zeta} = 0, \quad (84)$$

$$\frac{d \ln T_v}{dt} - \kappa \frac{d \ln p}{dt} = 0, \quad (85)$$

$$R_d T_v + \frac{p}{\pi} \frac{\partial \Phi}{\partial \log \pi} = 0, \quad (86)$$

$$\frac{d\phi}{dt} - gw = 0, \quad (87)$$

$$1 + \mu - \frac{p}{\pi} \frac{\partial \log p}{\partial \log \pi} = 0, \quad (88)$$

$$\ln \pi = \zeta + Bs. \quad (89)$$

4.11 GEM

The equations of GEM model (Girard et al., 2014) written out for a log-hydrostatic-pressure like coordinate ζ system co-rotating with the earth are:

with horizontal velocity \mathbf{V}_h , generalized vertical velocity $\dot{\zeta} = \frac{d\zeta}{dt}$, vertical velocity w , geopotential ϕ , virtual temperature T_v , pressure p and hydrostatic pressure π . The variable μ is a measure of departure from hydrostatic balance, s is related to the surface pressure π_s and B is a metric term. The last two variables are defined in (79).

4.12 FVM

- 5 The fully compressible Euler equations solved in FVM are given as

$$\frac{\partial \mathcal{G}\rho_d}{\partial t} + \nabla \cdot (\mathbf{v}\mathcal{G}\rho_d) = 0, \quad (90a)$$

$$\frac{\partial \mathcal{G}\rho_d \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{v}\mathcal{G}\rho_d \mathbf{u}) = \mathcal{G}\rho_d \left(-\Theta \tilde{\mathbf{G}} \nabla \phi' - \frac{\mathbf{g}}{\theta_a} (\theta' + \theta_a(\epsilon q'_v - q_c - q_r)) - \mathbf{f} \times \left(\mathbf{u} - \frac{\theta}{\theta_a} \mathbf{u}_a \right) + \mathbf{M} \right), \quad (90b)$$

$$\frac{\partial \mathcal{G}\rho_d \theta'}{\partial t} + \nabla \cdot (\mathbf{v}\mathcal{G}\rho_d \theta') = \mathcal{G}\rho_d \left(-\tilde{\mathbf{G}}^T \mathbf{u} \cdot \nabla \theta_a - \frac{L}{c_p \pi} \left(\frac{\Delta q_{vs}}{\Delta t} + E_r \right) + \mathcal{H} \right), \quad (90c)$$

$$\phi' = c_p \theta_0 \left[\left(\frac{R_d}{p_0} \rho_d \theta (1 + q_v/\varepsilon) \right)^{R_d/c_v} - \pi_a \right], \quad (90d)$$

- 10 which describe the conservation laws of dry mass (90a), momentum (90b), and dry entropy (90c). Dependent variables in (90) are dry density ρ_d , three-dimensional physical velocity vector \mathbf{u} , potential temperature perturbation θ' , and Exner pressure perturbation π' , with the thermodynamic variables related by the gas law (90d)¹. Mixing ratios of water vapor, cloud condensate, and rain are denoted as q_v , q_c , and q_r , respectively. All primed variables correspond to deviations from an ambient state (denoted by subscript "a") that satisfies a balanced subset of (90), thus $\psi' = \psi - \psi_a$, where $\psi = u, v, w, \theta, \dots$; see ? and ?. The
15 subscript "0" appearing with θ_0 refers to a constant reference value. Symbols appearing on the rhs of the momentum equation (90b) are the coefficient

$$\Theta := \frac{\theta (1 + q_v/\varepsilon)}{\theta_0 (1 + q_t)} \quad (91)$$

- in front of the pressure gradient term with Θ/θ_0 the density potential temperature, the gravity vector $\mathbf{g} \equiv (0, 0, -g)$ ², the Coriolis parameter \mathbf{f} , and $\varepsilon = 1/\varepsilon - 1$ with $\varepsilon = R_d/R_v$. The governing equations (90) are formulated with respect to a geospherical
20 coordinate system and a generalised height-based terrain-following vertical coordinate³. Associated symbols are the Jacobian of the metric tensor \mathcal{G} , a matrix of metric coefficients $\tilde{\mathbf{G}}$, its transpose $\tilde{\mathbf{G}}^T$, and the transformation of the physical to the contravariant velocity $\mathbf{v} = \tilde{\mathbf{G}}^T \mathbf{u}$; see ? and ? for discussion. The symbol \mathbf{M} in (90b) subsumes metric forces due to the curvature of the sphere (?) and momentum dissipation, whereas \mathcal{H} in (90c) represents the diffusion of heat.

5 Diffusion, Stabilization, Filters and Fixers

- 25 Most dynamical cores implement specialized techniques for diffusion or stabilization. Diffusion is a numerical technique that removes spurious numerical noise from the simulation, where the numerical noise typically arises because of inaccuracies

¹Note that ϕ' represents a normalised Exner pressure perturbation.

²In the shallow- versus deep-atmosphere form of the governing equations, gravity is constant $g \equiv g_c$ or varies with height as $g = g_c (a/r)^2$, respectively.

³For simplicity, the vertical coordinate is assumed to be time-independent in the current presentation.

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, or wave-mode 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 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 summarized here:

For damping, in the corrector step a 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.

$$F_d(\mathbf{v}) = -f_d \bar{a}_c^{-2} \nabla \tilde{\nabla} \cdot \left(\nabla \left(\tilde{\nabla} \cdot \mathbf{v} + \frac{\Delta}{\Delta z} \left(w - \overline{w}_{cc}^c \right)^i \right) \right). \quad (92)$$

f_d typically attains values between $\frac{1}{1000\Delta t}$ and $\frac{1}{250\Delta t}$, and \bar{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 $F_{D4}(v_n)$.

$$F_{D2}(v_n) = 4K_h \tilde{\nabla}^2(v_n) \quad (93)$$

$$F_{D2}(\theta) = a_c \tilde{\nabla} \cdot \left(K_h \frac{\Delta\theta}{\Delta n} \right). \quad (94)$$

$$F_{D4}(v_n) = -k_4 a_e^2 \tilde{\nabla}^2(\tilde{\nabla}^2(v_n)). \quad (95)$$

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

$$F_D(w) = -k_w a_c^2 \nabla^2(\nabla^2(w)) \quad (96)$$

5.2 FV3

Explicit dissipation in FV³ 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. FV³ 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 monotonicity constraint has been disabled, 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. This dissipation is of the same order of accuracy as the divergence damping; however the nondimensional coefficient needed is much smaller, usually by at least 3–4 times, than the divergence damping. Both divergence damping and hyperdiffusion are applied along the Lagrangian surfaces and are recomputed every acoustic timestep; there is no explicit damping perpendicular to the Lagrangian surfaces. An option to convert kinetic energy lost to the hyperdiffusion to heat is available. FV³ can also apply a horizontal Smagorinsky eddy diffusion if desired.

A substantial amount of wave absorption is provided by the flexible-lid (constant-pressure) upper boundary; FV³ 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.

FV³ 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.

5.3 GEM

Via an application of (??), scalar viscosity is employed for wind components and tracers. A vertical sponge layer, also via an application of (??), is employed on wind components and T_v with a vertical modulation on the 6 top levels and a maximum damping coefficient of $.380000 \times 10^6 \text{ m}^2/\text{s}$ at the top. For stabilization purpose, the temporal discretization of GEM presented in section 6.7 uses an off-centering parameter $b^A = 0.6$.

In GEM model, shape-preserving solutions are computed for tracers by using the quasi-monotone semi-Lagrangian (QMSL) method (Bermejo and Staniforth, 1992).

5.4 Tempest

Repeated application of the scalar and vector viscosity operators

6 Temporal Discretizations

[ALL] Describe the time-stepping scheme / temporal discretization employed by your dynamical core here.

6.1 Runge-Kutta

6.1.1 Ullrich-Kinnmark-Gray 5 step 3rd order scheme

- 5 Explicit terms are evolved using a Runge-Kutta method which supports a large stability bound for spatial discretizations with purely imaginary eigenvalues. This particular scheme is based on Kinnmark and Gray (1984a, b) and takes the form

$$\begin{aligned} \psi^{(1)} &= \psi^{(0)} + \frac{\Delta t}{5} f(\psi^{(0)}), \\ \psi^{(2)} &= \psi^{(0)} + \frac{\Delta t}{5} f(\psi^{(1)}), \\ \psi^{(3)} &= \psi^{(0)} + \frac{\Delta t}{3} f(\psi^{(2)}), \\ 10 \quad \psi^{(4)} &= \psi^{(0)} + \frac{2\Delta t}{3} f(\psi^{(3)}), \\ \psi^{(5)} &= -\frac{1}{4}\psi^{(0)} + \frac{5}{4}\psi^{(1)} + \frac{3\Delta t}{4} f(\psi^{(4)}). \end{aligned} \quad (97)$$

6.2 Semi-implicit time integration of the fully compressible Euler equations in FVM

- In the following we provide an outline of the semi-implicit time stepping scheme for the fully compressible Euler equations in FVM (Section 7.11). A comprehensive discussion of the integration scheme can be found in ?? for dry dynamics; and in ? and 15 ? for extensions to moist dynamics. The generic two-time-level second-order template algorithm employed in the integration is given as

$$\psi_{\mathbf{i}}^{n+1} = \mathcal{A}_{\mathbf{i}}(\tilde{\psi}^n, \mathbf{V}^{n+1/2}, (\mathcal{G}\rho_d)^n, (\mathcal{G}\rho_d)^{n+1}) + 0.5 \Delta t R^\psi|_{\mathbf{i}}^{n+1}, \quad \tilde{\psi}^n \equiv \psi^n + 0.5 \Delta t R^\psi|_i^n. \quad (98)$$

- In (98), ψ represents the solution variable, R^ψ is the respective rhs, and \mathcal{A} symbolises an advective transport operator assumed here to be the non-oscillatory finite-volume MPDATA (Multidimensional Positive Definite Advection Transport Algorithm) 20 scheme (??)⁴.

The integration of the system (90) 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 (98). Second, the thermodynamic (90c), momentum (90b), and moisture equations enter (98) with $\psi = u, v, w, \theta', \dots$, $\mathbf{V} \equiv \mathbf{v}\mathcal{G}\rho_d$, and the rhs R^ψ which is generally depending on all prognostic variables. A high degree of implicitness in the representation of the rhs forcings is achieved by inverting the overall discrete system 25 (98) to obtain closed-form expressions for the velocity updates – the procedure is facilitated by the co-located arrangement of variables on the computational mesh. Retained on the rhs of the derived closed-form velocity expressions is the pressure gradient term. The third step in the solution procedure is to formulate an implicit boundary value problem for the pressure variable ϕ' using an evolutionary form of the equation of state (90d). An $\mathcal{O}(\Delta t^2)$ integration of this equation with a Euler

⁴Furthermore, in (98) the vector index \mathbf{i} denotes the spatial position on the computational grid, Δt is the time step size between levels n and $n + 1$.

backward template algorithm in the spirit of (98) leads a Helmholtz equation. The associated 3D elliptic boundary value problem is solved iteratively using a bespoke preconditioned Generalised Conjugate Residual approach (??). Nonlinearities in R^ψ and the solution-dependent coefficients of the Helmholtz problem are lagged behind and executed in an outer iteration.

6.3 Forward-backward vertically-Lagrangian dynamics

5 FV³ 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 of the pressure-gradient force is evaluated backwards-in-time using the algorithm of Lin (1997); the nonhydrostatic component of the vertical pressure gradient force is evaluated using a semi-implicit solver. This forward-backward timestep is referred to as
10 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.

6.4 DYNAMICO

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
20 3-stage scheme ARK(2,3,2) is used (Giraldo et al., 2013).

6.5 ICON

The time stepping scheme of 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 ration of the maximum wind speed in the mesosphere, which is in part covered by the vertical domain, can be close
25 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 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).

6.6 Semi-Implicit time integration

6.7 The 2 time level semi-Lagrangian implicit time discretization in GEM model

Consider a frictionless adiabatic prognostic equation of the form:

$$\frac{dF}{dt} + G = 0, \quad (99)$$

- 5 where F represents one of the prognostic variables and G represents the remaining terms, some of which are non-linear. Based on the semi-Lagrangian scheme, The equation (99) is discretized as:

$$\frac{F^A - F^D}{\Delta t} + b^A G^A + (1 - b^A) G^D = 0, \quad (100)$$

with A (Arrival) = (\mathbf{r}, t) , D (Departure) = $(\mathbf{r} - \Delta\mathbf{r}, t - \Delta t)$ and \mathbf{r} is the 3D grid position. The displacements $\Delta\mathbf{r}$ are obtained by the iterative process:

$$10 \quad \Delta\mathbf{r}^i = [b^A \mathbf{V}(\mathbf{r}, t) + (1 - b^A) \mathbf{V}(\mathbf{r} - \Delta\mathbf{r}^{i-1}, t - \Delta t)] \Delta t, \quad (101)$$

where \mathbf{V} is the 3D wind vector. From (100), we regroup unknown terms on the left-hand side and known terms on the right-hand side, that is:

$$\frac{F^A}{\tau} + G^A = \frac{F^D}{\tau} - \beta G^D = R, \quad (102)$$

- where $\tau = \Delta t b^A$ and $\beta = (1 - b^A)/b^A$. Cubic Lagrange interpolation is used for upstream evaluations (F^D, G^D). Each left-hand side term $\frac{F^A}{\tau} + G^A$ is then split into a linear part L and a nonlinear residual part N :

$$\frac{F^A}{\tau} + G^A = L + N = R. \quad (103)$$

The linearization is done around a constant in time, horizontally homogeneous, reference state. This results in the non-linear problem:

$$L^i = R - N^{i-1}, \quad (104)$$

- 20 which is solved with two iterations. In each iteration, the linear system (104) is reduced to an Helmholtz problem for one composite variable. It is solved with a direct solver using the Schwarz-type domain decomposition method on Yin-Yang grid (Qaddouri et al., 2008). The composite variable solution is used to update the prognostic variables at time t . Our procedure involves another iterative process linked to the estimation of $\mathbf{V}(\mathbf{r}, t)$ in the displacement calculations (101). This results in redoing the sequence of operations (101)-(104). At each time step, the cubic Lagrange interpolation is used to update the static halo region of both panels of the Yin-Yang grid (Qaddouri and Lee, 2011).

7 Dynamical Cores

This section provide a short description of unique features or design specifications from several modern dynamical cores.

7.1 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 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. Time-differencing is based on the third-order Adams-Bashforth scheme. Horizontal diffusion is included in the form of a $\nabla_z^4()$ operator acting on the vorticity, divergence, potential temperature, and tracer.

7.2 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. The grid is an hexagonal C-grid in the horizontal and a Lorenz grid in the vertical. DYNAMICO's design uniquely combines:

- A representation of the prognostic and diagnostic fields following the ideas of discrete differential geometry (Dubos et al., 2015)
 - A novel Hamiltonian formulation of the equations of motion in non-Eulerian coordinates (Dubos and Tort, 2014) which is imitated at the discrete level using building blocks from the literature (Thuburn et al., 2009; Ringler et al., 2010), and yielding, to a certain extent, an energy-conserving spatial discretization
 - A 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).

7.3 FV Cubed (FV³)

The Geophysical Fluid Dynamics Laboratory FV³ model uses a fully finite-volume discretization of the fully-compressible nonhydrostatic Euler equations on the equiangular gnomonic cubed-sphere grid on a terrain-following Lagrangian vertical coordinate. The flow is entirely along the Lagrangian surfaces, allowing vertical transport to be represented implicitly without additional advection terms. Fluxes are computed using the Piecewise-Parabolic Method of Colella and Woodward (1984) with an optional monotonicity constraint; in nonhydrostatic applications the monotonicity constraint is used primarily for tracer transport. The discretization is on the C-D grid described by Lin and Rood (1997) which acts as a simplified Riemann solver: the D-grid winds are interpolated to the C-grid and then advanced by half of an acoustic timestep, giving time-centered winds that can then be used to compute the fluxes and advance the flux terms by a full acoustic timestep. 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 solver prognoses horizontal winds, in the native Gnomonic local coordinate; virtual potential temperature, which is 5 conserved by the adiabatic dynamics; mass, represented by the difference in hydrostatic pressure between the top and bottom of a grid cell; and tracer mass. The nonhydrostatic 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, vorticity is a 3D cell-mean value.

7.4 Finite-volume module of the Integrated Forecasting System

10 The finite-volume module (FVM) of the Integrated Forecasting System (IFS) is developed at ECMWF (?). 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 generalised, optionally time-dependent, terrain-following vertical coordinate based on height. A centred two-time-level semi-implicit integration scheme is employed with 3D implicit treatment of acoustic, buoyant, and rotational modes (?). The associated 3D Helmholtz problem is solved iteratively using 15 a bespoke preconditioned Generalised Conjugate Residual approach. The integration procedure uses the multidimensional flux-form Eulerian non-oscillatory MPDATA advection scheme (??). The horizontal spatial discretisation 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 ? for an exposition. In both the horizontal and the vertical discretisation, 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 20 grid (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 parametrisation and the 25 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. 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.

7.5 Global Environmental Multiscale (GEM) Model

30 In GEM model, we use the Yin-Yang grid and horizontal discretization is done on an Arakawa C grid. The vertical coordinate ζ is of a log-hydrostatic-pressure type and vertical discretization is based on the Charney-Phillips grid. A 2 time level semi-Lagrangian implicit time discretization is implemented as described in section 6.7. A scalar viscosity is employed for wind

components and tracers via an application of (??). Viscosity operations are applied after the completion of the dynamic time step.

7.6 High-Order Method Modeling Environment (HOMME)

HOMME stands for the High Order Method Modeling Environment which provides the framework for the CAM-SE and ACME-Atmosphere dynamical cores along with several experimental versions including a non-hydrostatic model. It was 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 by default in a cube-sphere structure, but variable-resolutions grids with conforming edges may also be employed. Each quadrilateral extends in the radial direction to form a column using the shallow-atmosphere approximation $r \approx a$ where a is the mean radius of the Earth. It employs a hybrid terrain-following / pressure coordinate $\eta \in (0, 1]$ which transforms smoothly from a pure pressure coordinate at the top of the model to a terrain-following surface-pressure coordinate at the bottom. A conventional vector-invariant form for the moist primitive equations is employed with prognostic zonal and meridional wind components u, v , temperature T , surface pressure p_s , and tracer mixing ratios q_i . The PDEs are discretized using a split approximation, with a nodal 4th order spectral element discretization in the horizontal and the mimetic (mass and energy conserving) 2nd order finite difference discretization of ? 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,p) placed on layer midpoints while vertical fluxes $m\dot{\eta}$ are placed on layer interfaces.

A floating Lagrange formulation is used for tracer advection and optionally for the vertical dynamics as well, in which vertical levels are advected with the fluid. After several time-steps, the levels are remapped onto the fixed Eulerian η grid in order to prevent pressure surfaces from crossing. Several limiter options are available including a sign-preserving limiter and a monotone optimization base limiter described in ?. Diffusion is applied in the horizontal with a 4th order hyperviscosity operator, decomposed into two applications of the Laplacian operator in divergence / vorticity form. The physics, tracer, dynamics, and vertical remapping schemes are applied in a time split manner with adjustable sub-cycling ratios. Tracer advection is discretized in time with a 3-stage 2nd order strong stability preserving (SSP) scheme of Spiteri and Ruuth ?. The dynamics are also discretized using one of several explicit Runge-Kutta time schemes. Physics-dynamics coupling may be achieved using a single adjustment at the end of the physics timestep, through tendency update every dynamics step, or by a hybrid of the two.

In addition to the primitive equation model, HOMME also hosts an experimental nonhydrostatic dynamical core which retains many of the features of the traditional CAM-SE model. This model makes use of the shallow-atmosphere Euler equations of motion in terrain-following η coordinates, as first described by ?. 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 , the surface pressure p_s , potential temperature θ , and geopotential height ϕ . Fields are currently co-located in the vertical and explicit time-stepping is used. Planned improvements call for vertically implicit time stepping, and possible Charney-Phillips staggering. This model has been successfully applied to the dry DCMIP-2012 dynamical core tests

including (vertical SE) tracer transport, orographic waves, gravity waves, and the dry baroclinic instability. Moist dynamics, physics/dynamics coupling, and implicit time integration routines are currently under development.

7.7 ICON

The ICON model 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. The prognostic variables are the normal wind v_n at the edge mid points of full levels, the vertical wind w in the circumcenters of the triangles on half levels and virtual potential temperature θ_v , full air density ρ , which includes moisture and hydrometeor densities and tracer mixing ratios q_x with respect to the full air density. The discretization in time employs a two time level predictor 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 order to avoid the reflection of gravity waves. The horizontal diffusion, which is applied at full model time steps, combines a flow dependent Smagorinski 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 neighbour communication only, thus allowing massive parallelization.

7.8 Model for Prediction Across Scales (MPAS)

[SKAMAROCK]

7.9 Naval Research Laboratory NEPTUNE Model

[VINNER, REINECKE]

25 7.10 Ocean-Land-Atmosphere Model (OLAM)

[WALKO]

7.11 Tempest

The Tempest model (Ullrich, 2014; Guerra and Ullrich, 2016) uses a horizontal spectral element discretization and vertical staggered nodal finite element method based on the cubed-sphere grid with terrain-following height-based coordinate. The

standard Eulerian equations are employed with moist density ρ , thermodynamic closure θ_v and tracer density ρq . These continuous equations are given in section 4.10.1. The implementation includes both fully explicit time integration, using the UKG53 scheme described in section 6.1.1, and implicit-explicit options, where horizontal terms are explicitly discretized and vertical terms are treated implicitly. Scalar hyperviscosity is employed for ρ , θ and tracer variables via repeated application of (??). Vector hyperviscosity is also applied by decomposing the horizontal vector Laplacian into divergence damping (??) and vorticity damping (??) terms. Both viscosity operations are applied after the completion of all Runge-Kutta sub-cycles.

7.12 Nonhydrostatic ICosahedral Atmospheric Model (NICAM) Model

[MIURA]

8 Idealized Physical Parameterizations

10 8.1 Kessler Physics

The cloud microphysics update according to the following equation set:

$$\frac{\Delta\theta}{\Delta t} = -\frac{L}{c_p\pi} \left(\frac{\Delta q_{vs}}{\Delta t} + E_r \right) \quad (105)$$

$$\frac{\Delta q_v}{\Delta t} = \frac{\Delta q_{vs}}{\Delta t} + E_r \quad (106)$$

$$\frac{\Delta q_c}{\Delta t} = -\frac{\Delta q_{vs}}{\Delta t} - A_r - C_r \quad (107)$$

$$15 \quad \frac{\Delta q_r}{\Delta t} = -E_r + A_r + C_r - V_r \frac{\partial q_r}{\partial z}, \quad (108)$$

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

$$p = \rho R_d T (1 + 0.61 q_v) \quad (109)$$

20 with p the pressure, ρ 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 Π equation.

$$\pi = \left(\frac{p}{p_0} \right)^{\frac{R_d T}{c_p}} \quad (110)$$

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

$$q_{vs}(p, T) = \left(\frac{380.0}{p} \right) \exp \left(17.27 \times \frac{T - 273.0}{T - 36.0} \right) \quad (111)$$

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

$$A_r = k_1(q_c - a) \quad (112)$$

$$C_r = k_2 q_c q_r^{0.875} \quad (113)$$

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

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

$$q_c^{n+1} = \max(q_c^r - \Delta q_r, 0) \quad (114)$$

$$q_r^{n+1} = \max(q_r^r - \Delta q_r + S, 0) \quad (115)$$

10 where S is the sedimentation term and Δq_r is defined as

$$\Delta q_r = q_c^n - \frac{q_c^n - \Delta t \max(A_r, 0)}{1 + \Delta t C_r} \quad (116)$$

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

$$E_r = \frac{1}{\rho} \frac{\left(1 - \frac{q_v}{q_{vs}}\right) C (\rho q_r)^{0.525}}{5.4 \times 10^5 + \frac{2.55 \times 10^6}{pq_{vs}}} \quad (117)$$

With ventilation factor C define as

$$15 \quad C_r = 1.6 + 124.9(\rho q_r)^{0.2046} \quad (118)$$

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

$$V_r = 36349(\rho q_r)^{0.1346} \left(\frac{\rho}{\rho_0}\right)^{-\frac{1}{2}} \quad (119)$$

9 Surface Fluxes on an Aqua-Planet with Prescribed Sea Surface Temperatures

20 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

$$\begin{aligned} C_d &= C_{d0} + C_{d1}|\mathbf{v}_a| && \text{for } |\mathbf{v}_a| < 20 \text{ m s}^{-1} \\ C_d &= 0.002 && \text{for } |\mathbf{v}_a| \geq 20 \text{ m s}^{-1}, \end{aligned} \quad (120)$$

where C_{d0} and C_{d1} are 7.0×10^{-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

$$5 \quad |\mathbf{v}_a| = \sqrt{u_a^2 + v_a^2}. \quad (121)$$

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

$$C_E = C_H = 0.0011. \quad (122)$$

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

$$10 \quad z_a = \frac{R_d T_{\nu,a}}{g} \frac{(\ln p_s - \ln p_-)}{2}, \quad (123)$$

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

15 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

$$\frac{\partial \mathbf{v}_a}{\partial t} = -\frac{C_d |\mathbf{v}_a| \mathbf{v}_a}{z_a} \quad (124)$$

$$\frac{\partial T_a}{\partial t} = \frac{C_H |\mathbf{v}_a| (T_s - T_a)}{z_a} \quad (125)$$

$$\frac{\partial q_a}{\partial t} = \frac{C_E |\mathbf{v}_a| (q_{sat,s} - q_a)}{z_a}. \quad (126)$$

20 We note that the wind at the surface is taken to be zero and therefore does not appear explicitly in (124). 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

$$q_{sat}(p) \approx \varepsilon \frac{e_s(T_s)}{p} \approx \frac{\varepsilon}{p} e_0^* e^{-(L/R_\nu)[(1/T_s) - (1/T_0)]}, \quad (127)$$

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

25 The final form of the surface fluxes will vary for models with other choices of prognostic variables. For example, if potential temperature Θ_a is used (125) takes the form

$$\frac{\partial \Theta_a}{\partial t} = \frac{C_H |\mathbf{v}_a| (T_s - T_a)}{z_a} \left(\frac{p_0}{p_a} \right)^{R_d/c_p} \quad (128)$$

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 (124), (128) and (126) would need to be multiplied by the density of the air ρ .

In order to ensure numerical stability, each of the aforementioned surface fluxes are applied via a semi-implicit operator.

- 5 We demonstrate this procedure on the temperature evolution equation (125). First, the time derivative is expanded using a backward Euler operator,

$$\frac{T_a^{n+1} - T_a^n}{\Delta t} = \frac{C_H |\mathbf{v}_a^n| (T_s - T_a^{n+1})}{z_a}. \quad (129)$$

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 . $|\mathbf{v}_a^n|$

- 10 is evaluated at the current time step and C_H is constant. The equation can now be solved for T_a^{n+1}

$$T_a^{n+1} = \frac{T_a^n + C_H |\mathbf{v}_a^n| T_s \frac{\Delta t}{z_a}}{1 + C_H |\mathbf{v}_a^n| \frac{\Delta t}{z_a}}. \quad (130)$$

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

$$\mathbf{v}_a^{n+1} = \frac{\mathbf{v}_a^n}{1 + C_d^n |\mathbf{v}_a^n| \frac{\Delta t}{z_a}} \quad (131)$$

$$q_a^{n+1} = \frac{q_a^n + C_E |\mathbf{v}_a^n| q_{sat,s}^n \frac{\Delta t}{z_a}}{1 + C_E |\mathbf{v}_a^n| \frac{\Delta t}{z_a}}, \quad (132)$$

- 15 with the time-level dependent coefficient C_d^n . Notice that the second term in the numerator of (131) is absent in the case of the zonal and meridional wind. This is because the wind is set to zero at the surface.

9.1 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

20 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

$$\frac{\partial u}{\partial t} = -\frac{1}{\rho} \frac{\partial \rho \overline{w' u'}}{\partial z} \quad (133)$$

$$\frac{\partial v}{\partial t} = -\frac{1}{\rho} \frac{\partial \rho \overline{w' v'}}{\partial z} \quad (134)$$

$$\frac{\partial q}{\partial t} = -\frac{1}{\rho} \frac{\partial \rho \overline{w' q'}}{\partial z}. \quad (135)$$

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

$$\frac{\partial \Theta}{\partial t} = -\frac{1}{\rho} \frac{\partial \rho \overline{w' \Theta'}}{\partial z}. \quad (136)$$

Here u' , v' , w' , Θ' 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 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

$$5 \quad \frac{\partial T}{\partial t} = -\frac{1}{\rho} \left(\frac{p}{p_0} \right)^\kappa \frac{\partial \rho \overline{w' \Theta'}}{\partial z}. \quad (137)$$

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 layers

$$\overline{w'u'} = -K_m \frac{\partial u}{\partial z} \quad (138)$$

$$10 \quad \overline{w'v'} = -K_m \frac{\partial v}{\partial z} \quad (139)$$

$$\overline{w'\Theta'} = -K_E \frac{\partial \Theta}{\partial z} \quad (140)$$

$$\overline{w'q'} = -K_E \frac{\partial q}{\partial z}. \quad (141)$$

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 15 surface flux calculated in Appendix 9 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

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

$$K_m = C_d |\mathbf{v}_a| z_a \exp \left(- \left[\frac{p_{top}-p}{p_{strato}} \right]^2 \right) \quad \text{for } p \leq p_{top}. \quad (142)$$

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

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

$$K_E = C_E |\mathbf{v}_a| z_a \exp \left(- \left[\frac{p_{top}-p}{p_{strato}} \right]^2 \right) \quad \text{for } p \leq p_{top}. \quad (143)$$

20 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 used in the model CAM which can serve as a template routine.

10 Conclusions

25 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
5 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,
10 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

- 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.
- Colella, P. and Woodward, P. R.: The Piecewise Parabolic Method (PPM) for Gas-Dynamical Simulations, *J. Comput. Phys.*, 54, 174–201, 5 1984.
- 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.
- Du, Q., Gunzburger, M. D., and Ju, L.: Constrained centroidal Voronoi tessellations for surfaces, *SIAM Journal on Scientific Computing*, 24, 1488–1506, 2003.
- 10 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-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 15 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.
- 20 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.
- 25 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.
- Harris, L. M., Lin, S.-J., and Tu, C.: High resolution climate simulations using GFDL HiRAM with a stretched global grid, *Journal of 30 Climate*, 2016.
- 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.
- Kageyama, A. and Sato, T.: The Yin-Yang grid: An overset grid in spherical geometry, *Geochem. Geophys. Geosyst.*, 5, doi:10.1029/2004GC000734, 2004.
- 35 Kessler, E.: On the distribution and continuity of water substance in atmospheric circulation, 1969.
- Kinnmark, I. P. and Gray, W. G.: One step integration methods with maximum stability regions, *Math. Comput. Simulat.*, 26, 87–92, doi:10.1016/0378-4754(84)90039-9, 1984a.

- Kinnmark, I. P. and Gray, W. G.: One step integration methods of third-fourth order accuracy with large hyperbolic stability limits, *Math. Comput. Simulat.*, 26, 181–188, doi:10.1016/0378-4754(84)90056-9, 1984b.
- 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., Dudhia, J., and Hassiotis, A. D.: An upper gravity-wave absorbing layer for NWP applications, *Monthly Weather Review*, 136, 5 3987–4004, doi:10.1175/2008MWR2596.1, 2008.
- 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, *Mon. Wea. Rev.*, 132, 2293–2307, 2004a.
- Lin, S.-J.: A ?vertically Lagrangian? finite-volume dynamical core for global models, *Monthly Weather Review*, 132, 2293–2307, 2004b.
- 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.
- 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 15 Wiley & Sons, 2009.
- 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.
- 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.
- 20 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.
- 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 25 tessellations, *Ocean Dynamics*, 58, 475–498, 2008.
- Ringler, T. D., Thuburn, J., Klemp, J. B., and Skamarock, W. C.: A unified approach to energy conservation and potential vorticity dynamics 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.
- 30 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.
- 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.
- 35 Soong, S.-T. and Ogura, Y.: A comparison between axisymmetric and slab-symmetric cumulus cloud models, *J. Atmos. Sci.*, 30, 879–893, 1973.
- 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.

Ullrich, P.: A global finite-element shallow-water model supporting continuous and discontinuous elements, *Geosci. Model Dev.*, 7, 3017–3035, 2014.

Zängl, G.: The ICON (ICOrahedral 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.