

## A Semi-Implicit Version of the MPAS-Atmosphere Dynamical Core

STEVEN SANDBACH,\* JOHN THUBURN, AND DANAIL VASSILEV<sup>†</sup>

*University of Exeter, Exeter, United Kingdom*

MICHAEL G. DUDA

*National Center for Atmospheric Research, Boulder, Colorado*

(Manuscript received 14 February 2015, in final form 29 May 2015)

### ABSTRACT

An important question for atmospheric modeling is the viability of semi-implicit time integration schemes on massively parallel computing architectures. Semi-implicit schemes can provide increased stability and accuracy. However, they require the solution of an elliptic problem at each time step, creating concerns about their parallel efficiency and scalability. Here, a semi-implicit (SI) version of the Model for Prediction Across Scales (MPAS) is developed and compared with the original model version, which uses a split Runge–Kutta (SRK3) time integration scheme. The SI scheme is based on a quasi-Newton iteration toward a Crank–Nicolson scheme. Each Newton iteration requires the solution of a Helmholtz problem; here, the Helmholtz problem is derived, and its solution using a geometric multigrid method is described. On two standard test cases, a midlatitude baroclinic wave and a small-planet nonhydrostatic gravity wave, the SI and SRK3 versions produce almost identical results. On the baroclinic wave test, the SI version can use somewhat larger time steps (about 60%) than the SRK3 version before losing stability. The SI version costs 10%–20% more per step than the SRK3 version, and the weak and strong scalability characteristics of the two versions are very similar for the processor configurations the authors have been able to test (up to 1920 processors). Because of the spatial discretization of the pressure gradient in the lowest model layer, the SI version becomes unstable in the presence of realistic orography. Some further work will be needed to demonstrate the viability of the SI scheme in this case.

### 1. Introduction

The use of a semi-implicit time integration scheme to handle the fast waves in atmospheric models was first introduced to enable large time steps to be taken without loss of stability (Robert 1969; Robert et al. 1972; Bourke 1974; Hoskins and Simmons 1975). Semi-implicit schemes require the solution of a Helmholtz problem at least once per time step, but, provided this can be done efficiently, the longer time steps allowed can lead to an overall gain in model efficiency compared to an explicit time integration

scheme. Early applications of semi-implicit schemes treated only certain linearized dynamical terms implicitly. Later, it was shown (Cullen 2001; Cullen and Salmond 2003) that a predictor–corrector scheme that iterates toward a more fully implicit scheme, including implicit treatment of nonlinear dynamical terms and even physical parameterizations, could lead to improved accuracy and better representation of balances between different processes. Preoperational testing of the new Met Office dynamical core Even Newer Dynamics for General Atmospheric Modelling of the Environment (ENDGame; Wood et al. 2014) showed that a more fully implicit scheme conferred greater stability and robustness, thereby allowing a reduction in the artificial damping and diffusion used to stabilize the model, further improving accuracy (Walters et al. 2014).

Recently, the desire for parallel scalability on massively parallel computing architectures has reinvigorated interest in the use of quasi-uniform spherical grids for atmospheric modeling, in order to avoid the communications bottleneck that arises from the polar

---

\* Current affiliation: Met Office, Exeter, United Kingdom.

<sup>†</sup> Current affiliation: Cobham Technical Services, Oxfordshire, United Kingdom.

---

*Corresponding author address:* John Thurn, Department of Mathematics, University of Exeter, North Park Road, Exeter EX4 4QF, United Kingdom.  
E-mail: j.thurn@exeter.ac.uk

resolution clustering on the longitude–latitude grid. Several global atmospheric models have recently been developed on quasi-uniform grids specifically for such computer architectures (Sato et al. 2008; Walko and Avissar 2008; Qaddouri and Lee 2011; Ullrich and Jablonowski 2012a; Skamarock et al. 2012; Zängl et al. 2015). However, because of concerns about whether a three-dimensional Helmholtz problem could be solved in an efficient and scalable way, almost all of these models retain an implicit time integration scheme only for the vertical propagation of information, combined with some form of explicit time integration scheme for the horizontal propagation of information. Such schemes are known as horizontally explicit vertically implicit (HEVI). Although such schemes are certainly viable, they do typically require some damping of acoustic waves to ensure numerical stability over the desired parameter range: for example, in the form of divergence damping or off-centering (e.g., Sato et al. 2008; Walko and Avissar 2008; Skamarock et al. 2012; Zängl et al. 2015, and references therein). It would be valuable to know whether parallel scalability issues do indeed make semi-implicit time stepping schemes uncompetitive or whether they might, in fact, remain viable or even advantageous given a suitable Helmholtz solver.

Even more recently, it has been shown that Helmholtz problems and Poisson problems of the sort arising in atmospheric modeling can be solved efficiently and with good parallel scalability using geometric multigrid methods (Heikes et al. 2013; Müller and Scheichl 2014; Dedner et al. 2015, manuscript submitted to *Int. J. Numer. Methods Fluids*). See, for example, Fulton et al. (1986) for a clear introduction to multigrid methods in the context of atmospheric modeling. Such methods require only local (rather than global) data communication at each smoother iteration. Moreover, the conditioning of the Helmholtz problem depends on the horizontal acoustic wave Courant number  $c_s \Delta t / \Delta x$ , where  $c_s$  is the sound speed,  $\Delta t$  is the time step, and  $\Delta x$  is the horizontal grid spacing. In practice, on a quasi-uniform grid, the time step is reduced in proportion to  $\Delta x$  as resolution is increased, so the horizontal acoustic wave Courant number remains bounded (typically less than 10), and the Helmholtz problem does not become worse conditioned at higher resolution. Consequently, (i) a shallow multigrid hierarchy is sufficient (see section 4), and (ii) the number of V-cycles and smoother iterations required does not depend strongly on resolution. These considerations motivate us to develop a semi-implicit version of an existing atmospheric model designed for massively parallel computing architectures and to compare its performance and parallel scalability to the original HEVI time stepping version.

The model in question is the Model for Prediction Across Scales–Atmosphere (MPAS–Atmosphere). It is described in detail by Skamarock et al. (2012) and references therein. Its main features are the following. It solves the compressible nonhydrostatic equations. The horizontal grid is a spherical centroidal Voronoi tessellation with a C-grid placement of variables. A general terrain-following vertical coordinate is used with a Lorenz-grid staggering of the vertical velocity relative to other variables. The spatial discretization uses a combination of finite difference and finite volume ideas; it conserves mass, mass-weighted potential temperature, and tracers and respects hydrostatic and geostrophic balance. The original time integration scheme is the three-stage Runge–Kutta split explicit (or, more precisely, split HEVI) scheme (SRK3) described by Wicker and Skamarock (2002). Each stage of the Runge–Kutta scheme is broken down into a number of substeps in which the time tendencies are updated using the fast acoustic and gravity wave terms in the equations. The substeps use a forward–backward time integration scheme in which the vertical coupling terms are treated implicitly.

In this paper, we replace the original time integration scheme by a semi-implicit (SI) one. Various factors were considered in the choice of SI scheme. It is desirable to keep the spatial discretization unchanged and to retain a single-step time integration scheme, both to facilitate a clean comparison between the SI and SRK3 schemes and to avoid major structural changes to the code (see Fig. 3 below). As noted above, early semi-implicit schemes for atmospheric models treated only certain linear terms implicitly. Linearly implicit schemes, such as Runge–Kutta–Rosenbrock (RKR) schemes, originally described in the ODE literature, are becoming more widely applied in complex models for the solution of PDEs (e.g., Kar 2006; John et al. 2006; Ullrich and Jablonowski 2012b). However, John et al. (2006) found RKR schemes to be 3–4 times more expensive than a Crank–Nicolson scheme for their test cases, because the RKR linear problem must be solved accurately to ensure accuracy of the scheme overall. Also, we carried out some initial experiments with a Strang carryover scheme (Ullrich and Jablonowski 2012b), a simple variant of RKR, but found that adding “slow” and “fast” time step contributions separately led to unacceptably large imbalances. Finally, we were motivated by the results of Cullen (2001) and Cullen and Salmond (2003) mentioned above, along with a belief that the (weakly) nonlinear problem arising from a Crank–Nicolson time step could be solved for a cost comparable to that of the corresponding linearized problem (see section 2b below). Thus, we chose to implement and test a scheme

based on an iteration toward a Crank–Nicolson scheme. It is similar, in some respects, to the time scheme used in the Canadian Meteorological Centre’s Global Environment Multiscale (GEM) model (Yeh et al. 2002) and in ENDGame (Wood et al. 2014), though with Eulerian rather than semi-Lagrangian time derivatives.

The developments described in this paper are based on version 2.0 of the MPAS-Atmosphere code. The main releases of the MPAS code are available online (<https://github.com/MPAS-Dev/MPAS-Release/releases>). The semi-implicit version described here is not yet part of a main release, but interested readers can obtain the code and some additional instructions for use online as well (<https://github.com/mgduda/MPAS-Release/releases/tag/v2.0-semi-implicit>).

Section 2 describes the formulation of the new time integration scheme and how this leads to a Helmholtz problem. A geometric multigrid solver is used to solve the Helmholtz problem; the multigrid structure and related operators are described in section 3, and the Helmholtz solver itself is described in section 4. The structure and communications costs of the SRK3 and SI algorithms are compared in section 5. Some sample results and discussion of performance and parallel scalability are presented in section 6.

## 2. Formulation

### a. Continuous equations

The continuous governing equations for the MPAS-Atmosphere are given by Skamarock et al. (2012). Here, we summarize them briefly [see Skamarock et al. (2012) for a full discussion].

A general terrain-following vertical coordinate  $\zeta$  is used such that height  $z$  is given by

$$z = \zeta + A(\zeta)h_s(\mathbf{x}_H, \zeta), \quad (1)$$

where  $\mathbf{x}_H$  is the horizontal position, and  $h_s(\mathbf{x}_H, 0)$  is the surface height. Let  $\nabla\zeta = (\zeta_H, \zeta_z)$ , where  $\zeta_H = \nabla_H\zeta$  is the horizontal gradient of  $\zeta$  at constant height. Also, define  $\mathbf{z}_H = -\zeta_H/\zeta_z$ , the slope of the coordinate surfaces; these quantities are used in computing the divergence and the pressure gradient terms below.

The prognostic equations are written in terms of flux variables:

$$(\mathbf{V}, \Theta_m, Q_j) = \tilde{\rho}_d(\mathbf{v}, \theta_m, q_j). \quad (2)$$

Here,  $\tilde{\rho}_d = \rho_d/\zeta_z$  where  $\rho_d$  is the density of dry air, and  $\mathbf{v} = (\mathbf{v}_H, w)$  is the velocity vector with horizontal and vertical components  $\mathbf{v}_H$  and  $w$ . The  $q_j$  are the mixing ratios of various water species. A modified moist potential temperature

$$\theta_m = \theta(1 + q_v R_v/R_d) \quad (3)$$

is used, where  $q_v$  is the water vapor mixing ratio, and  $R_v$  and  $R_d$  are the gas constants for water vapor and dry air, respectively.

Define  $\Omega = \mathbf{V} \cdot \nabla\zeta$  to be the component of the mass flux normal to  $\zeta$  surfaces, and let  $\mathbf{k}$  be the vertical unit vector. Then the governing equations may be written as follows:

$$\begin{aligned} \frac{\partial \mathbf{V}_H}{\partial t} = & -\frac{\rho_d}{\rho_m} \frac{1}{\zeta_z} \left( \nabla_\zeta p - \mathbf{z}_H \frac{\partial p}{\partial z} \right) - \eta \mathbf{k} \times \mathbf{V}_H \\ & - \mathbf{v}_H \nabla_\zeta \cdot \mathbf{V} - \frac{\partial \Omega \mathbf{v}_H}{\partial \zeta} - \tilde{\rho}_d \nabla_\zeta K \\ & + W \mathbf{k} \times \Omega_e - \frac{\mathbf{v}_H W}{r_e} + \mathbf{F}_{\mathbf{v}_H}, \end{aligned} \quad (4)$$

$$\begin{aligned} \frac{\partial W}{\partial t} + rW = & -\frac{\rho_d}{\rho_m} \left( \frac{\partial p}{\partial \zeta} + g \tilde{\rho}_m \right) - (\mathbf{V} \cdot \mathbf{v}W)_\zeta \\ & + \frac{uU + vV}{r_e} - \mathbf{V} \cdot (\mathbf{k} \times \Omega_e) + F_W, \end{aligned} \quad (5)$$

$$\frac{\partial \Theta_m}{\partial t} = -(\mathbf{V} \cdot \nabla \theta_m)_\zeta + F_{\Theta_m}, \quad (6)$$

$$\frac{\partial \tilde{\rho}_d}{\partial t} = -(\mathbf{V} \cdot \nabla)_\zeta, \quad \text{and} \quad (7)$$

$$\frac{\partial Q_j}{\partial t} = -(\mathbf{V} \cdot \nabla q_j)_\zeta + F_{Q_j}. \quad (8)$$

The pressure  $p$  is obtained via the equation of state:

$$p = p_0 \left( \frac{R_d \zeta_z \Theta_m}{p_0} \right)^\gamma, \quad (9)$$

where  $p_0$  is a constant reference pressure, and  $\gamma$  is the ratio of specific heat capacities at constant pressure and constant volume  $\gamma = c_p/c_v$ . The density of moist air  $\rho_m$  is given by

$$\frac{\rho_m}{\rho_d} = 1 + q_v + q_c + q_r + \dots, \quad (10)$$

with  $q_v, q_c, q_r$ , et cetera, the mixing ratios of water vapor, cloud water, rainwater, and so on. Following Klemp et al. (2008), a linear damping term  $rW$ , with  $r$  a function of altitude, is included in the  $W$  equation to provide a mechanism for damping waves near the model top. The other variables not yet defined are the gravitational acceleration  $g$ ; absolute vertical vorticity  $\eta$ ; horizontal kinetic energy  $K = |\mathbf{v}_H|^2/2$ ; Earth’s rotation vector  $\Omega_e$ ; Earth’s radius  $r_e$ ; and  $\mathbf{F}_{\mathbf{v}_H}, F_W, F_{\Theta_m}$ , and  $F_{Q_j}$ , which represent source terms. Finally,  $\nabla_\zeta$  is the horizontal gradient along  $\zeta$  surfaces, and it is convenient to express the three-dimensional divergence of the flux of any scalar  $b$  as

$$\frac{1}{\zeta_z} \nabla \cdot (\rho_d \mathbf{v}b) = (\nabla \cdot \mathbf{V}b)_\zeta \equiv \nabla_\zeta \cdot (\mathbf{V}_H b) + \frac{\partial \Omega b}{\partial \zeta}, \quad (11)$$

where  $\nabla_\zeta$  is the horizontal divergence operator along a  $\zeta$  surface.

Note that the horizontal pressure gradient term in (4) is written in an equivalent but slightly different form from Skamarock et al. (2012). The form used here more closely reflects how the term is discretized in the MPAS code and also facilitates the derivation of the Helmholtz problem below. Note, also, that the pressure gradient terms and buoyancy term in (4) and (5) are actually evaluated in terms of departures from hydrostatically balanced reference thermodynamic profiles that are functions only of  $z$ , as in Klemp et al. (2007); this reduces truncation errors in the calculation of the horizontal pressure gradient, where coordinate surfaces are sloping.

*b. Time discretization*

The overarching idea is to use a Crank–Nicolson time discretization for the dynamical equations, which should give excellent stability even for long time steps. However, the Crank–Nicolson scheme is only second-order accurate and so might lead to dispersion errors for advected quantities, such as the moisture variables. Therefore, we retain the third-order Runge–Kutta time integration scheme of Wicker and Skamarock (2002) for the advection of moisture variables; this, however, necessitates a mild approximation in the evaluation of  $\rho_m$  (see section 2e).

Introduce the notation  $\mathbf{T}_{v_H}$ ,  $T_W$ ,  $T_{\Theta_m}$ , and  $T_\rho$  (the tendencies) as shorthand for the right-hand sides of (4), (5), (6), and (7), respectively. A Crank–Nicolson time discretization of (4)–(7) is then

$$\frac{\mathbf{V}_H^{n+1} - \mathbf{V}_H^n}{\Delta t} = \langle \mathbf{T}_{v_H} \rangle, \quad (12)$$

$$\frac{W^{n+1} - W^n}{\Delta t} + rW^{n+1} = \langle T_W \rangle, \quad (13)$$

$$\frac{\Theta_m^{n+1} - \Theta_m^n}{\Delta t} = \langle T_{\Theta} \rangle, \quad \text{and} \quad (14)$$

$$\frac{\tilde{\rho}_d^{n+1} - \tilde{\rho}_d^n}{\Delta t} = \langle T_\rho \rangle, \quad (15)$$

where  $\Delta t$  is the time step, superscripts  $n$  and  $n + 1$  indicate fields at the current and future time levels, and

$$\langle T \rangle = \alpha T^{n+1} + \beta T^n. \quad (16)$$

Following Klemp et al. (2008), the  $W$  damping term uses a backward-in-time discretization. For consistency  $\alpha + \beta = 1$ , and for stability  $\alpha \geq 0.5$ . The usual Crank–Nicolson scheme has  $\alpha = 0.5$ , and we use this value for all results presented below, except where stated in

sections 6b and 6c. A choice of  $\alpha > 0.5$  may be used in situations where it is desirable to damp fast waves.

Equations (12)–(15) represent our target time discretization. However, the unknown fields at time level  $n + 1$  appear on both sides of each equation, with spatial coupling through derivative terms (and some interpolation/averaging) and, in most cases, nonlinearly. Thus, we have a coupled nonlinear system of equations to solve at each time step.

The system is solved iteratively using an approximate Newton method. There are numerous variants of approximate Newton methods (Knoll and Keyes 2004), including quasi-Newton methods, in which the Jacobian matrix is approximated (Martínez 2000); inexact Newton methods, in which the linear system for the Newton update is solved only approximately (Dembo et al. 1982; Jay 2000); and simplified Newton methods, in which the Jacobian is not updated during the Newton iterations. Our scheme involves all of these approximations, but for brevity we will refer to it as a quasi-Newton method. The terms retained in the Jacobian [the left-hand sides of (26)–(29) below] are those that describe acoustic and gravity waves for linear perturbations about some reference thermodynamic profiles. These are the stiffest terms and are the ones that are crucial for the convergence of the Newton iterations.

Let superscript ( $l$ ) indicate the best available estimate for each field at step  $n + 1$  after  $l$  iterations. After  $l$  iterations (12)–(15) will not be satisfied exactly, but will have some residuals defined by

$$\mathbf{V}_H^{(l)} - \mathbf{V}_H^n - \Delta t \langle \mathbf{T}_{v_H} \rangle^{(l)} = -\mathbf{R}_{v_H}, \quad (17)$$

$$(1 + r\Delta t)W^{(l)} - W^n - \Delta t \langle T_W \rangle^{(l)} = -R_W, \quad (18)$$

$$\Theta_m^{(l)} - \Theta_m^n - \Delta t \langle T_{\Theta} \rangle^{(l)} = -R_{\Theta}, \quad \text{and} \quad (19)$$

$$\tilde{\rho}_d^{(l)} - \tilde{\rho}_d^n - \Delta t \langle T_\rho \rangle^{(l)} = -R_\rho, \quad (20)$$

where

$$\langle T \rangle^{(l)} = \alpha T^{(l)} + \beta T^n. \quad (21)$$

Now seek increments  $\mathbf{V}'_H$ ,  $W'$ ,  $\Theta'_m$ ,  $\tilde{\rho}'_d$  to the prognostic fields

$$\mathbf{V}_H^{(l+1)} = \mathbf{V}_H^{(l)} + \mathbf{V}'_H, \quad (22)$$

$$W^{(l+1)} = W^{(l)} + W', \quad (23)$$

$$\Theta_m^{(l+1)} = \Theta_m^{(l)} + \Theta'_m, \quad \text{and} \quad (24)$$

$$\tilde{\rho}_d^{(l+1)} = \tilde{\rho}_d^{(l)} + \tilde{\rho}'_d, \quad (25)$$

designed to reduce the residuals:

$$\mathbf{V}'_H + \alpha\Delta t \frac{1}{\zeta_z} \nabla_\zeta p' = \mathbf{R}_{\mathbf{V}_H}, \quad (26)$$

$$W' + \alpha\Delta t \left( \frac{\partial p'}{\partial \zeta} + \overline{g\tilde{\rho}'_d} \right) = R_W, \quad (27)$$

$$\Theta'_m + \alpha\Delta t \left[ \theta^*(\hat{\mathbf{V}} \cdot \mathbf{V}')_\zeta + \overline{W'\zeta_z \frac{\partial \theta^*}{\partial \zeta}} \right] = R_\Theta, \quad \text{and} \quad (28)$$

$$\tilde{\rho}'_d + \alpha\Delta t (\hat{\mathbf{V}} \cdot \mathbf{V}')_\zeta = R_\rho. \quad (29)$$

Here,  $p'$  is related to  $\Theta'_m$  by the linearized equation of state:

$$\frac{1}{\gamma} \frac{p'}{p^*} = \frac{\Theta'_m}{\Theta^*}. \quad (30)$$

Asterisks indicate reference thermodynamic fields. In general, they may be functions of all three spatial coordinates as well as time and should not be confused with the reference profiles introduced by Klemp et al. (2007). They are assumed to satisfy the equation of state but are not required to be in hydrostatic balance. In the current implementation, they are set equal to the corresponding time level  $n$  fields. An approximate divergence operator has been introduced, defined by

$$(\hat{\mathbf{V}} \cdot \mathbf{V}')_\zeta = \nabla_\zeta \cdot (\mathbf{V}'_H) + \frac{\partial(\zeta_z W')}{\partial \zeta}. \quad (31)$$

The choice of retained terms in (28) merits further comment. It is motivated by a desire for a potential temperature increment satisfying

$$\theta'_m + \alpha\Delta t w' \frac{\partial \theta^*}{\partial z} = R_\theta, \quad (32)$$

in order to make the static stability  $N^2 = (g/\theta^*)\partial\theta^*/\partial z$  appear and hence to capture the gravity wave restoring mechanism in the approximate Jacobian. The resulting Helmholtz equation is then analogous to that for the ENDGame scheme of Wood et al. (2014). Combining (32) with (29) gives an equation for the increment to the density-weighted potential temperature [(28)].

The neglected terms on the left-hand sides of (26)–(29) include Coriolis terms, nonlinear advection terms, and the effect of the slope of the coordinate surfaces in the horizontal pressure gradient and in converting between  $W$  and  $\Omega$ . Scaling analysis confirms that the first two should indeed be negligible, because the relevant dimensionless parameters  $\alpha\Delta t|\Omega_e|$  and  $\alpha\Delta t|\mathbf{V}\mathbf{v}|$  will be small in practice. However, the effect of the slope of the coordinate surfaces can become important when the vertical resolution is much finer than the horizontal (see section 6d). Note that, at convergence, all the residuals go to zero, and we do solve the full system [(12)–(15)],

whatever approximations are made on the left-hand sides of (26)–(29).

To keep the notation concise, we have not made the spatial discretization explicit, except in one specific aspect: the overline indicates two terms that must be vertically averaged or interpolated because of the use of the Lorenz vertical grid staggering. This averaging has consequences for the form of the Helmholtz problem derived in the next section.

### c. Helmholtz problem

We now have a linear system [(26)–(30)] to be solved at each quasi-Newton iteration, but it is still spatially coupled and still involves several unknown fields. In this section, the system is reduced to a Helmholtz equation for the single unknown field  $p'$ .

First use (30) to eliminate  $\Theta'_m$ :

$$\frac{1}{\gamma} \frac{\tilde{\rho}'_d \theta^*}{p^*} p' + \alpha\Delta t \left[ \theta^*(\hat{\mathbf{V}} \cdot \mathbf{V}')_\zeta + \overline{W'\zeta_z \frac{\partial \theta^*}{\partial \zeta}} \right] = R_\Theta, \quad (33)$$

or

$$\frac{1}{\gamma} \frac{\tilde{\rho}'_d}{p^*} p' + \alpha\Delta t \left[ (\hat{\mathbf{V}} \cdot \mathbf{V}')_\zeta + \frac{N^2}{g} W' \right] = \frac{R_\Theta}{\theta^*}, \quad (34)$$

where

$$N^2 = \frac{g}{\theta^*} \zeta_z \frac{\partial \theta^*}{\partial \zeta}. \quad (35)$$

If we define the vertical operator  $\mathcal{D}_2$  by

$$\mathcal{D}_2(X) = \frac{\partial(\zeta_z X)}{\partial \zeta} + \frac{N^2}{g} X, \quad (36)$$

then (34) may be written

$$\frac{1}{\gamma} \frac{\tilde{\rho}'_d}{p^*} p' + \alpha\Delta t [\nabla_\zeta \cdot (\mathbf{V}'_H) + \mathcal{D}_2(W')] = \frac{R_\Theta}{\theta^*}. \quad (37)$$

Next eliminate the divergence term by taking (34) minus (29):

$$\frac{1}{\gamma} \frac{\tilde{\rho}'_d}{p^*} p' - \tilde{\rho}'_d + \alpha\Delta t \overline{W' \frac{N^2}{g}} = \frac{R_\Theta}{\theta^*} - R_\rho. \quad (38)$$

Then use (38) to eliminate  $\tilde{\rho}'_d$  from (27):

$$\begin{aligned} & (1 + r\Delta t)W' + \alpha^2 \Delta t^2 \overline{W' N^2} + \alpha\Delta t \left( \frac{\partial p'}{\partial \zeta} + \frac{g}{\gamma} \frac{\tilde{\rho}'_d}{p^*} p' \right) \\ & = R_W + g\alpha\Delta t \left( \frac{R_\Theta}{\theta^*} - R_\rho \right), \end{aligned} \quad (39)$$

or, more compactly,

$$\mathcal{N}(W') + \alpha \Delta t \mathcal{D}_1(p') = R_{WP}, \quad (40)$$

where

$$\mathcal{N}(Y) = (1 + r \Delta t)Y + \alpha^2 \Delta t^2 \overline{\overline{N^2 Y}}, \quad (41)$$

$$\mathcal{D}_1(Z) = \frac{\partial Z}{\partial \zeta} + \frac{g}{\gamma p^*} \frac{\tilde{\rho}_d^*}{p^*} Z, \quad (42)$$

and

$$R_{WP} = R_W + g \alpha \Delta t \left( \frac{R_\Theta}{\theta^*} - R_\rho \right). \quad (43)$$

Finally, use (26) and (40) to eliminate  $\mathbf{V}'_H$  and  $W'$  from (37) to obtain the Helmholtz equation:

$$\mathcal{H}(p') = R_H, \quad (44)$$

where

$$\mathcal{H}(p') = \frac{1}{\zeta c_s^2} p' - \alpha^2 \Delta t^2 \left\{ \mathbf{V}'_\zeta \cdot \left[ \frac{1}{\zeta} \mathbf{V}'_\zeta(p') \right] + \mathcal{D}_2 \mathcal{N}^{-1}[\mathcal{D}_1(p')] \right\} \quad (45)$$

and

$$R_H = \frac{R_\Theta}{\theta^*} - \alpha \Delta t \mathcal{D}_2 \mathcal{N}^{-1}(R_{WP}) - \alpha \Delta t \mathbf{V}'_\zeta \cdot (R_{\mathbf{V}'_H}). \quad (46)$$

We have used  $c_s^2 = \gamma RT^*$  and the fact that reference thermodynamic fields satisfy the equation of state to write the coefficient of  $p'$  in terms of the reference sound speed  $c_s$ .

Boundary conditions are needed to close the Helmholtz problem. The appropriate conditions are that  $W'$  should vanish at the bottom and top boundaries. To take these into account, the coefficients for the  $\mathcal{D}_2$  and  $\mathcal{N}$  operators are set so as to ignore contributions from the bottom and top boundaries. The Helmholtz problem for  $p'$  then comprises the same number of discrete equations as unknowns, with the appropriate boundary conditions on  $p'$  accounted for implicitly.

An interesting feature of this Helmholtz problem, which arises from the use of the Lorenz vertical grid staggering, is the appearance of  $\mathcal{N}^{-1}$  on both the left- and right-hand sides of (44). Its appearance on the right-hand side means that a tridiagonal system of equations must be solved in each grid column in order to compute the right-hand side. Its appearance on the left-hand side slightly complicates the application of the smoother (see section 4). A tridiagonal system must also be solved at

the back-substitution stage (see section 2d). In contrast, for a Charney–Phillips vertical staggering, the analog of  $\mathcal{N}$  is simply a multiplicative factor, and  $\mathcal{N}^{-1}$  is trivial to evaluate; then no tridiagonal systems need to be solved to evaluate the right-hand side of the Helmholtz problem or in the back substitution, and the smoother is slightly simpler.

#### d. Back substitution

Having solved (44) to find  $p'$ , the increments to the prognostic variables  $\Theta'_m$ ,  $\mathbf{V}'_H$ , and  $W'$  are found by back substitution into (30), (26), and (40), respectively. Back substitution for  $W'$  requires the solution of a tridiagonal system to invert  $\mathcal{N}$ . The density increment  $\tilde{\rho}'_d$  is obtained from

$$\tilde{\rho}'_d + \alpha \Delta t (\mathbf{V} \cdot \mathbf{V}')_\zeta = R_\rho, \quad (47)$$

using the full divergence operator rather than the approximate version in (29). The advantage of doing this, rather than using (29) or the alternative (38), is that the mass continuity [(15)] is then satisfied exactly, ensuring local mass conservation and mass-tracer consistency (section 2e), even if the quasi-Newton iterations have not converged.

Having obtained the increments by back substitution, the estimates for the time level  $n + 1$  fields are updated [(22)–(25)].

The first-guess values for the time level  $n + 1$  fields are given by the time level  $n$  fields:

$$\mathbf{V}'_H^{(0)} = \mathbf{V}^n_H, \quad (48)$$

$$W^{(0)} = W^n, \quad (49)$$

$$\Theta_m^{(0)} = \Theta_m^n, \quad \text{and} \quad (50)$$

$$\tilde{\rho}_d^{(0)} = \tilde{\rho}_d^n. \quad (51)$$

#### e. Time discretization of moisture advection equations

Advection of moisture variables uses the third-order Runge–Kutta time scheme, as in Wicker and Skamarock (2002):

$$\tilde{\rho}_d^{*n} = \tilde{\rho}_d^n - \frac{\Delta t}{3} (\mathbf{V} \cdot \langle \mathbf{V} \rangle)_\zeta,$$

$$\tilde{\rho}_d^{*n} q^{*n} = \tilde{\rho}_d^n q^n - \frac{\Delta t}{3} (\mathbf{V} \cdot \mathbf{F}^n)_\zeta,$$

$$\tilde{\rho}_d^{**n} = \tilde{\rho}_d^n - \frac{\Delta t}{2} (\mathbf{V} \cdot \langle \mathbf{V} \rangle)_\zeta,$$

$$\tilde{\rho}_d^{**n} q^{**n} = \tilde{\rho}_d^n q^n - \frac{\Delta t}{2} (\mathbf{V} \cdot \mathbf{F}^n)_\zeta,$$

$$\tilde{\rho}_d^{n+1} = \tilde{\rho}_d^n - \Delta t (\mathbf{V} \cdot \langle \mathbf{V} \rangle)_\zeta, \quad \text{and}$$

$$\tilde{\rho}_d^{n+1} q^{n+1} = \tilde{\rho}_d^n q^n - \Delta t (\mathbf{V} \cdot \mathbf{F}^{**n})_\zeta, \quad (52)$$

where the tracer fluxes  $\mathbf{F}^n$ ,  $\mathbf{F}^*$ , and  $\mathbf{F}^{**}$  are evaluated using the latest available mixing ratios  $q^n$ ,  $q^*$ , and  $q^{**}$ , respectively, but in each case using the time-averaged mass fluxes  $\langle \mathbf{V} \rangle$ .

For the results presented below, the quasi-Newton iterations to update  $\mathbf{V}_H$ ,  $W$ ,  $\Theta_m$ , and  $\tilde{\rho}_d$  are carried out first and then the advection of moisture variables. This results in an approximation to the Crank–Nicolson time integration scheme: namely, that  $\mathbf{T}_{\mathbf{V}_H}^{n+1}$  and  $T_W^{n+1}$  are evaluated using moisture values from time level  $n$  rather than  $n+1$ . This approximation may be expected to have only a very small effect on the results, because the tendencies depend only weakly on the moisture values, and the moisture values will usually not vary dramatically over one time step. The code has, in fact, been written to allow a moisture advection update at every quasi-Newton iteration [with the last two lines in (52) replaced by

$$\begin{aligned} \tilde{\rho}_d^{(l+1)} &= \tilde{\rho}_d^n - \Delta t (\nabla \cdot \langle \mathbf{V} \rangle^{(l+1)})_{\zeta}; \\ \tilde{\rho}_d^{(l+1)} q^{(l+1)} &= \tilde{\rho}_d^n q^n - \Delta t (\nabla \cdot \mathbf{F}^{**})_{\zeta} \end{aligned} \quad (53)$$

and using the latest available estimate of the time average mass flux  $\langle \mathbf{V} \rangle^{(l+1)}$ . This has allowed us to compare the approximate Crank–Nicolson scheme with the full Crank–Nicolson scheme and confirm that the differences are indeed negligible. Since the full scheme is significantly more expensive, because the evaluation of the advective fluxes is relatively expensive, the approximate scheme is used for all the results shown below.

An important property for a scalar transport scheme is mass-tracer consistency: the mass fluxes used to advect scalars should be identical to those used to update the density; otherwise it is not possible to ensure that advection conserves tracer mass and, at the same time, preserves a constant tracer mixing ratio (e.g., Jöckel et al. 2001; Wong et al. 2013, and references therein). The scalar advection scheme implemented in MPAS has the mass-tracer consistency property, provided the density and mass fluxes satisfy the equation on the penultimate line of (52). Since only a small number of quasi-Newton iterations will be taken in practice, we therefore require

$$\tilde{\rho}_d^{(l)} = \tilde{\rho}_d^n - \Delta t (\nabla \cdot \langle \mathbf{V} \rangle^{(l)})_{\zeta}. \quad (54)$$

Now, (54) clearly does not hold for the first-guess  $l=0$ , because  $\tilde{\rho}_d^{(0)} = \tilde{\rho}_d^n$ . However, for subsequent iterations

$$\langle \mathbf{V} \rangle^{(l+1)} = \langle \mathbf{V} \rangle^{(l)} + \alpha \mathbf{V}', \quad (55)$$

and

$$\tilde{\rho}_d' = -\alpha \Delta t (\nabla \cdot \mathbf{V}')_{\zeta} + R_{\rho}, \quad (56)$$

where

$$R_{\rho} = -(\tilde{\rho}_d^{(l)} - \tilde{\rho}_d^n) - \Delta t (\nabla \cdot \langle \mathbf{V} \rangle^{(l)})_{\zeta}. \quad (57)$$

Adding (56) and (57) gives (54), as required, for any  $l > 0$ . Note that the mass-tracer consistency property is obtained irrespective of how accurately the Helmholtz problem is solved or how well converged the quasi-Newton iterations are. But note, also, that it does depend on using (47) rather than (29) to obtain density increments consistent with the mass flux increments.

### 3. Multigrid grid structure

A suitably nested hierarchy of grids is needed for the multigrid solver described in section 4. In fact, such a grid hierarchy is a natural by-product of the grid generation tool used to generate the MPAS grids, which uses a recursive subdivision strategy. We simply need to save the coarser-grid information rather than discarding it.

Figure 1 illustrates the relationship between the cells on a fine grid and those on the next coarser grid. A subset of the fine cells is centered on the coarse cells, while the remaining fine cells straddle the edges of the coarse cells.

Restriction and prolongation operators are needed to transfer fields from a fine grid to the next coarser grid and from a coarse grid to the next finer grid, respectively. For the restriction operator, an area-weighted average is used: for example,

$$A_i^{(c)} p_i^{(c)} = \sum_j w_{ij} A_j^{(f)} p_j^{(f)}, \quad (58)$$

where  $A_i^{(c)}$  is the area of the  $i$ th coarse cell,  $A_j^{(f)}$  is the area of the  $j$ th fine cell, and  $p_i^{(c)}$  and  $p_j^{(f)}$  are the corresponding values of the variable to be restricted. We have found that, on quasi-uniform (i.e., unstretched) grids, a simple choice of weights is sufficiently accurate:  $w_{ij} = 1$  when fine cell  $i$  is centered on coarse cell  $j$ ,  $w_{ij} = 1/2$  when fine cell  $i$  straddles an edge of coarse cell  $j$ , and  $w_{ij} = 0$  otherwise.<sup>1</sup> The prolongation operator is given by a simple sampling/interpolation:

$$p_j^{(f)} = \sum_i w_{ij} p_i^{(c)}, \quad (59)$$

<sup>1</sup>MPAS can use more general grids in which the density of grid cells varies, providing local refinement (Skamarock et al. 2012). In this case, the definition of the grid hierarchy and the restriction and prolongation operators for the implicit version becomes more complicated; this extension will be discussed elsewhere.

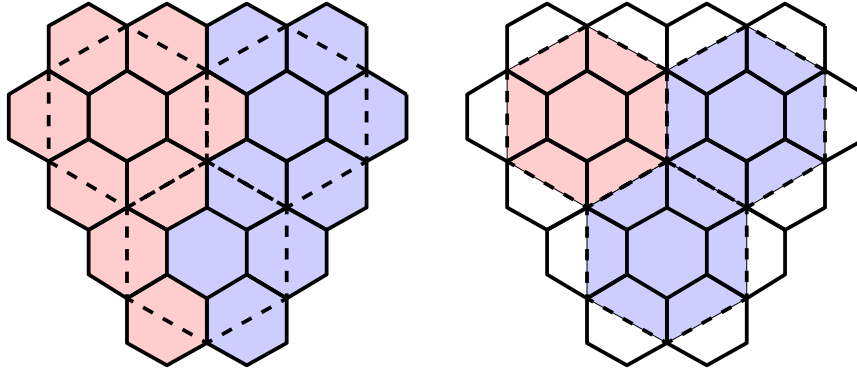


FIG. 1. Schematic showing the relationship between coarse grid cells (dashed) and fine grid cells (solid) for two adjacent grids in the grid hierarchy. Both panels show the same region of cells. The cells are colored according to which subdomain owns (left) the fine cells and (right) the coarse cells.

with the same  $w_{ij}$  as above. Note that restriction and prolongation operators are needed only for cell-based quantities, not for edge-based quantities.

To run the model on multiple processors the domain is decomposed into a number of subdomains. The domain decomposition is precomputed and stored in a graph file. The same graph file may be used for the finest grid in the multigrid hierarchy as is used for the single (fine) grid in the SRK3 model version. However, in the multigrid case, we must also decide which grid subdomain owns coarser grid cells. We make the simple choice that a coarse cell belongs to the same subdomain as the fine cell at its center. This choice is applied recursively down the hierarchy. Figure 1 shows that both restriction and prolongation operations require information from neighboring subdomains; a one-cell-deep layer, or “halo,” of data surrounding each subdomain must be exchanged before each restriction or prolongation. (This is a disadvantage of the hexagonal grid; on a quadrilateral or triangular grid, it is possible to choose the grid hierarchy and decomposition such that restriction and prolongation operations do not need a halo exchange.)

The MPAS-Atmosphere software uses Fortran-derived data types, called blocks, each block containing all the data pertaining to its region of the domain, and using pointers to the next or previous block to form a linked list. This linked list concept provides a convenient framework that can be extended to include multiple resolution grids using pointers to the next coarser and finer grids (Fig. 2).

#### 4. Helmholtz solver

The Helmholtz problem [(44)] is solved using a geometric multigrid method (e.g., Fulton et al. 1986). The

grid is coarsened only in the horizontal direction. The geometrical relation between fine and coarse grids and the restriction and prolongation operators for mapping between them are described in section 3 above. A single V-cycle is used. On the finest grid, a number of iterations of some relaxation scheme (see below) are taken to relax  $p'$  toward the solution of the Helmholtz problem. Then the residual in the Helmholtz problem is calculated and restricted to the next coarser grid, where it serves as the right-hand side in a Helmholtz problem for a correction to  $p'$ . A number of relaxation iterations are taken on this grid, and the coarsening process is repeated down to some desired depth. After some relaxation iterations on the coarsest grid, the solution is prolonged to the next finer grid and added as a correction to the solution previously obtained on that grid. The relaxation and prolongation process is repeated until the finest grid is reached, and some final relaxation iterations are taken on the finest grid.

The smoother involves a Jacobi iteration in the horizontal and a line solve in the vertical. To be explicit, write the horizontal Laplacian part of the Helmholtz operator at level  $k$  in column  $i$  as

$$\left\{ \nabla_{\zeta} \cdot \left[ \frac{1}{\zeta_z} \nabla_{\zeta} (p') \right] \right\}_{k,i} = \frac{1}{A_i} \sum_e \frac{l_e}{d_e \zeta_{zk,e}} (p'_{k,i'} - p'_{k,i}), \quad (60)$$

where the sum is over the edges  $e$  of column  $i$ , column  $i'$  is the neighbor of column  $i$  across edge  $e$ ,  $l_e$  is the length of edge  $e$ ,  $d_e$  is the distance between the centers of columns  $i$  and  $i'$ ,  $\zeta_{zk,e}$  is the value of  $\zeta_z$  at level  $k$  averaged from cells  $i$  and  $i'$  to edge  $e$ , and  $A_i$  is the horizontal area of the base of column  $i$  (Skamarock et al. 2012). The Helmholtz problem [(44)] becomes



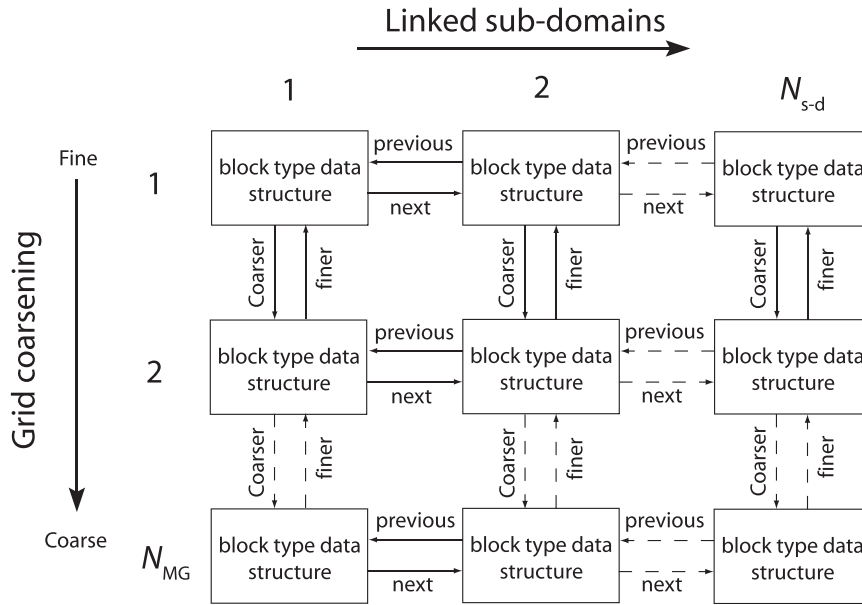


FIG. 2. Schematic showing how the MPAS data structure has been extended to include the hierarchy of grids needed for a multigrid method.

$$\begin{aligned} & \left( \frac{1}{\xi_z c_s^2} + \alpha^2 \Delta t^2 \frac{1}{A_i} \sum_e \frac{l_e}{d_e \xi_{zk,e}} \right) p'_{k,i} \\ & - \alpha^2 \Delta t^2 \{ \mathcal{D}_2 \mathcal{N}^{-1} [ \mathcal{D}_1 (p') ] \}_{k,i} \\ & = (R_H)_{k,i} + \alpha^2 \Delta t^2 \frac{1}{A_i} \sum_e \frac{l_e}{d_e \xi_{zk,e}} p'_{k,i'} \end{aligned} \quad (61)$$

where

$$C_{k,i} = \left( \frac{1}{\xi_z c_s^2} + \alpha^2 \Delta t^2 \frac{1}{A_i} \sum_e \frac{l_e}{d_e \xi_{zk,e}} \right)_{k,i} \quad (65)$$

and

$$(R_q)_{k,i} = (R_H)_{k,i} + \alpha^2 \Delta t^2 \frac{1}{A_i} \sum_e \frac{l_e}{d_e \xi_{zk,e}} p_{k,i'}^{(m)} \quad (66)$$

Then a smoother iteration is defined by simultaneously updating all  $p'$  in column  $i$  to satisfy (61) while holding  $p'$  in neighboring columns at their previous values:

$$\begin{aligned} & \left( \frac{1}{\xi_z c_s^2} + \alpha^2 \Delta t^2 \frac{1}{A_i} \sum_e \frac{l_e}{d_e \xi_{zk,e}} \right) p_{k,i}^{(m+1)} \\ & - \alpha^2 \Delta t^2 \{ \mathcal{D}_2 \mathcal{N}^{-1} [ \mathcal{D}_1 (p^{(m+1)}) ] \}_{k,i} \\ & = (R_H)_{k,i} + \alpha^2 \Delta t^2 \frac{1}{A_i} \sum_e \frac{l_e}{d_e \xi_{zk,e}} p_{k,i'}^{(m)}, \end{aligned} \quad (62)$$

where  $p^{(m)}$  is the estimate for  $p'$  after  $m$  smoother iterations.

The calculation of  $p^{(m+1)}$  is made more complicated by the appearance of  $\mathcal{N}^{-1}$  on the left of (62). It is convenient to define  $q^{(m+1)}$  by

$$[ \mathcal{N} (q^{(m+1)}) ]_{k+1/2,i} = [ \mathcal{D}_1 (p^{(m+1)}) ]_{k+1/2,i} \quad (63)$$

and hence write (62) as

$$C_{k,i} p_{k,i}^{(m+1)} - \alpha^2 \Delta t^2 \{ \mathcal{D}_2 [ q^{(m+1)} ] \}_{k,i} = (R_q)_{k,i}, \quad (64)$$

Next, eliminate  $p^{(m+1)}$  to obtain a tridiagonal system for  $q^{(m+1)}$ :

$$\mathcal{N} [ q^{(m+1)} ] - \alpha^2 \Delta t^2 \mathcal{D}_1 \mathcal{C}^{-1} \mathcal{D}_2 [ q^{(m+1)} ] = \mathcal{D}_1 \mathcal{C}^{-1} (R_q). \quad (67)$$

Having solved this system for  $q^{(m+1)}$ ,  $p^{(m+1)}$  is then found by back substitution in (64).

As an aside, the linear system arising from the vertically implicit acoustic substeps in the SRK3 scheme is solved by eliminating the pressure to leave a tridiagonal system for the vertical velocity (Klemp et al. 2007); in this way, the above complication of inverting  $\mathcal{N}$  is avoided. However, for the three-dimensional linear system of the SI scheme, eliminating pressure to leave an equation for the vertical velocity would lead to great complications, because  $\mathcal{D}_1$  does not commute with the horizontal Laplacian.

Alternatives to the horizontal Jacobi smoother that converge faster are possible, such as coloring schemes, which use the latest available results from neighboring

columns (e.g., Zhou and Fulton 2009). We have found the convergence rate of Jacobi to be adequate. Moreover, it has the advantage that results are independent of the order in which columns are updated; it thus presents no barrier to bit reproducibility when runs are repeated, even on different processor configurations.

On quadrilateral grids, Jacobi smoothers are typically used with underrelaxation. However, the analysis of Zhou and Fulton (2009) concludes that an underrelaxation parameter close to 1 (i.e., little or no underrelaxation) is optimal on a regular hexagonal grid, and our own numerical experimentation confirms that this remains true on a hexagonal–icosahedral spherical grid. Therefore, no underrelaxation is used with the Jacobi smoother.

An important characteristic of the Helmholtz operator is that it has an intrinsic horizontal length scale  $\alpha\Delta t c_s$ . If the horizontal grid spacing is comparable to or greater than  $\alpha\Delta t c_s$ —in other words, if the horizontal acoustic Courant number is less than about 1—then the Helmholtz operator is dominated by the contributions from column  $i$ , and the smoother iterations converge quickly.<sup>2</sup> Thus, once the multigrid solver has coarsened to this scale, few smoother iterations are needed, and it is not necessary to coarsen further. For typical flow and model parameters, we found that three multigrid levels (i.e., the original finest grid plus two levels of coarsening) were sufficient; using more levels gave no benefit, but the solver convergence deteriorated with fewer levels.

The fact that only a small number of multigrid levels are needed simplifies the computational implementation of the multigrid solver. For a Poisson problem (e.g., Heikes et al. 2013), a deeper multigrid hierarchy is needed. On coarser grids, this could result in very few grid columns per processor so that processors run out of work and communication costs dominate. To avoid this problem, computational subdomains must be merged on the coarser grids. For the Helmholtz problem, in contrast, a shallow hierarchy is sufficient, and no subdomain merging is necessary.

Because the Helmholtz problem is embedded within an outer quasi-Newton iteration, it is not necessary to solve the Helmholtz problem to a tight tolerance. It is only necessary to solve it to sufficient accuracy to avoid harming the convergence of the quasi-Newton iteration. Solving it to a higher accuracy would increase the

computational cost for no benefit. After some experimentation, our preferred configuration is to take a single V-cycle, with one smoother iteration on the descending branch, two smoother iterations on the ascending branch, and four smoother iterations on the coarsest grid. This is enough to reduce the residual in the Helmholtz problem by several orders of magnitude. (We have also experimented with a full multigrid method, which involves a growing sequence of V-cycles starting at the coarsest grid; however, this was significantly more expensive, while giving no noticeable benefit.)

## 5. Comparison of algorithms and communication load

An overview of the SRK3 and SI algorithms is shown in Fig. 3. The work flow and data flow for the two algorithms is remarkably similar, which has greatly facilitated the development of the SI version. In particular, the SI version requires no special treatment at the first time step [in contrast, for example, to a Strang carryover scheme; Ullrich and Jablonowski (2012b)], and no extra fields need to be saved to restart the model.

For the SRK3 scheme, the Runge–Kutta loop is executed three times, once per stage. In code segment A, the dynamical tendencies are calculated and added to the physical tendencies (excluding fast microphysics) that were calculated outside the loop. Next, in code segment B, the acoustic substepping loop is executed (following Klemp et al. 2007); this involves converting prognostic variables to perturbations and taking the required number of acoustic substeps. By default, there are 1, 3, and 6 acoustic substeps on the first, second, and third Runge–Kutta stages, respectively, giving 10 acoustic substeps in total. In code segment C, perturbation variables are converted back to full model variables, and some diagnostic quantities are computed. Finally, in the last two code segments, advective fluxes are computed and used to update moisture variables (D), and some further diagnostic quantities are computed (E).

The SI solver follows a similar structure. The number of outer quasi-Newton iterations may be chosen by the user; we have used three. Code segments A and E are the same as in the SRK3 scheme. Code segment C is largely the same as for SRK3 but performs only a subset of the calculations. The biggest difference is in code segment B, where the acoustic substepping is replaced by the Helmholtz solver. The Helmholtz solver requires (i) setting up the coefficients of the Helmholtz equation [(44)] (at the first iteration only), (ii) building the Helmholtz right-hand side [(46)], (iii) solving the Helmholtz problem using the multigrid solver (section 3), and (iv) back substitution to obtain the updated prognostic

<sup>2</sup>Note that, despite this horizontal decoupling, the Helmholtz problem remains well posed. Even in the limit of complete horizontal decoupling, the solution of the Helmholtz problem is unique; there is no undetermined “constant of integration” that could lead to large errors in horizontal gradients of  $p'$ .

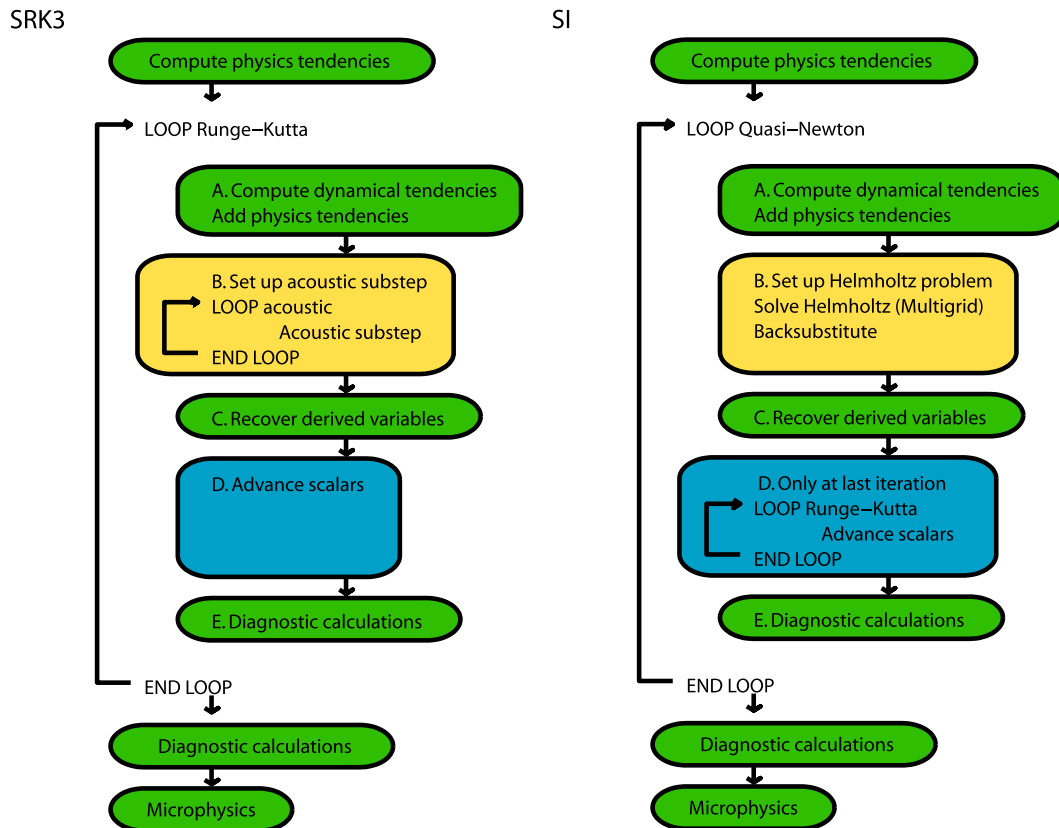


FIG. 3. Overview of the (left) SRK3 and (right) SI solver algorithms summarizing similarities and differences.

fields (section 2d). Code segment D is modified to include a Runge–Kutta loop for the moisture variable advection; by default, this is only executed on the final quasi-Newton iteration, though the user can choose other options. Table 1 summarizes these similarities and differences.

For parallel computation, at various stages in the computation, each subdomain needs information from its neighbors: a halo region of cells or edges surrounding the subdomain is filled with data by passing “messages” between processors. The cost of this communication can be significant or even dominant on large numbers of processors. Table 1 gives estimates of the size of messages passed by different code segments during the dynamical step. Define a message size of one unit

to be the amount of data involved in exchanging a single layer of halo cells for a cell-based variable, such as density. In some cases, a double layer of halo cells is exchanged; this corresponds to approximately two units. For an edge-based variable such as horizontal velocity, up to three halo layers may need to be exchanged. The innermost layer corresponds to one unit of data, and the second and third correspond to three units of data each. On the coarser grids used by the multigrid solver, the message size for a halo exchange decreases by a factor of about 0.5 per level of grid coarsening. Counting in this way, the total message size per time step is 178 units for SRK3 and 202 units for SI (assuming three quasi-Newton iterations). The communications load for the

TABLE 1. Summary of differences in algorithm and communications between SRK3 and SI. The message size is normalized, taking the total message size for one SRK3 step to be 100%.

| Code segment | Message size SRK3 | Message size SI | Comments  |
|--------------|-------------------|-----------------|---|
| A            | 0%                | 0%              | Same routine  |
| B            | 27.5%             | 46.6%           | Acoustic substeps vs Helmholtz solver                 |
| C            | 0%                | 0%              | SI uses a subset of the calculations in SRK3          |
| D            | 23.6%             | 24.7%           | SI does some additional computations compared to SRK3 |
| E            | 0%                | 0%              | Same routine  |
| Other        | 48.9%             | 42.1%           | One halo exchange saved in SI                         |

code segments A–E is given in the table, normalized by taking the total load for SRK3 to be 100%. Some additional halo exchanges occur during the time step but outside these code segments, bringing the total to 100% for SRK3. Code segment B has different communication patterns for SRK3 and SI, amounting to an additional 19% for SI. However, we were able to reduce the size of one halo exchange elsewhere, thereby saving about 7%. Thus, the SI code involves an overall increase in message size of approximately 12% per time step.

The communications cost of the Helmholtz solver is of particular interest, since such solvers are widely perceived to be expensive. A set of nine (cell based) coefficient fields are defined on the finest grid; these are then restricted to the required coarser grids, each restriction operation requiring a single-layer halo exchange. In the current implementation, this coefficient setup stage is done once per time step, though it could probably be done less frequently. (A reviewer has suggested an alternative, which is to restrict only the fields needed to compute the Helmholtz coefficients— $\theta^*$  and  $\tilde{\rho}_a^*/p^*$  plus some time-independent grid information that only needs to be restricted once at the start of the integration—and to compute the Helmholtz coefficients directly on the coarser grids; this could be cheaper than our current algorithm if the saving in communication outweighs the extra computation.) Then, during the course of the V-cycle, each restriction or prolongation operation and each smoother iteration requires a single-layer halo exchange. As noted above, the message size for a halo exchange decreases by a factor of about 0.5 per level of grid coarsening. The total communications cost to set up the Helmholtz coefficients and solve three times (once per Newton iteration) is 41 units. For the SI scheme, 81% of the message size for code segment B is associated with the finest grid; thus, the coarser-grid halo exchanges contribute relatively little to the communications burden.

## 6. Results

### a. Baroclinic instability test

The baroclinic instability test case of Jablonowski and Williamson (2006) was carried out with the SRK3 and SI versions of the model. For both versions, a horizontal resolution of 240 km was used (10 242 grid cells) with 41 nonuniformly spaced levels up to a model top at 45 km. A time step of 1800 s was used.

Figure 4 shows the surface pressure and the temperature at 850 hPa at day 9 produced by the SI scheme. The results from the SRK3 scheme appear identical by eye, so the figure also shows the differences between the

results for the two schemes. The wave appears to be very slightly more developed with the SI scheme, but only by a fraction of a hectopascal in the surface pressure and about 1.5 K in the 850-hPa temperature. The two schemes also give almost identical results for a passive tracer initialized with a moisture-like distribution (not shown). Finally, we verified that a passive tracer initialized with a constant mixing ratio retains that constant mixing ratio identically through the integration, confirming the mass-tracer consistency property (section 2e).

### b. Nonhydrostatic gravity wave test

To test the SI scheme in a nonhydrostatic regime, test case 3.1 of the Dynamical Core Model Intercomparison Project (DCMIP) suite (Ullrich et al. 2012) was carried out. The test comprises a basic state in balanced solid body rotation with a zonal velocity of  $20 \text{ m s}^{-1}$  on the equator, to which a horizontally localized but deep potential temperature perturbation is added. Deep gravity waves are generated, which radiate away from the initial perturbation with a maximum phase speed of about  $30 \text{ m s}^{-1}$  relative to the background flow. The radius of the planet is reduced (relative to Earth) by a factor of 125 so that the gravity wave wavelength is short enough for nonhydrostatic effects to be significant. The domain is 10 km deep, and uniform 1-km vertical grid spacing was used. A horizontal grid of 40 962 cells was used, corresponding to a horizontal grid length of about 1 km. A time step of 12 s was used, with 8 acoustic substeps for the SRK3 scheme.

Figure 5 shows the potential temperature perturbation along the equator after 3600 s from the SI scheme for  $\alpha = 0.5$  and  $\alpha = 0.55$ . The results are similar to those from the SRK3 scheme and from other models for which results are available. The figure also shows the differences between results from the SI and SRK3 schemes. When the SI scheme is centered ( $\alpha = 0.5$ ) the differences are extremely small, showing only a very slight phase lag (of order 3% of a wavelength) for the shortest waves. Since the SI scheme artificially reduces the frequency of high-frequency waves, this phase lag is exactly as expected theoretically. For the off-centered SI scheme ( $\alpha = 0.55$ ) the differences show some small but noticeable damping of the wave. Again, this is expected theoretically.

### c. Stability limit

Given the good stability properties of implicit time integration schemes, it is reasonable to ask whether the SI scheme might be able to run stably with longer time steps or with weaker artificial damping than the SRK3 scheme. The baroclinic wave test case of section 6a was

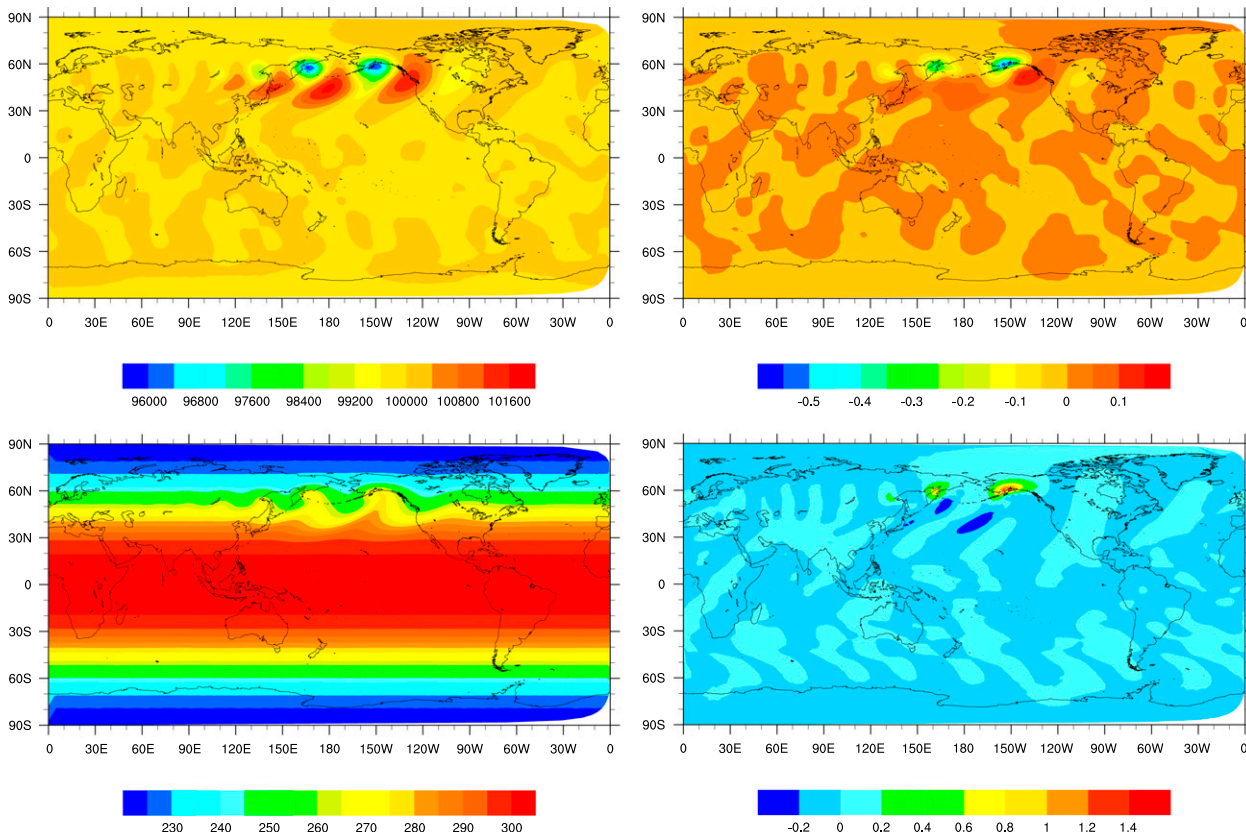


FIG. 4. Results at day 9 from the baroclinic wave test case: (top) surface pressure; (bottom) temperature at 850 hPa; (left) SI time integration scheme; (right) SI minus SRK3.

repeated for both model versions to determine the largest time step that permitted a stable 12-day integration. For these tests, neither model version used the  $W$  damping:  $r = 0$  in (5). The SRK3 version used the default values of the divergence damping coefficient  $\beta_d = 0.1$  and the off-centering coefficient  $\beta_s = 0.1$  (Klemp et al. 2007). The value  $\beta_s = 0.1$  in the SRK3 version corresponds to  $\alpha = 0.55$  in the SI version, with the difference that off-centering is applied only to acoustic wave terms in the SRK3 version but to all terms in the SI version. For the SI version, we tested both three and four quasi-Newton iterations; the cost of the extra Newton iteration might be justified if it provided a sufficient gain in stability.

Table 2 summarizes the empirical stability limits for the various configurations tested. With no off-centering, the SI version, even with four quasi-Newton iterations, is somewhat less stable than the default SRK3 configuration. However, with a modest amount of off-centering the SI version becomes more stable than the default SRK3 configuration, allowing time steps about 60% longer for  $\alpha = 0.55$  with three quasi-Newton iterations. In the centered case  $\alpha = 0.5$ , an extra Newton iteration

allows an increase of about 40% in the time step, which is more than sufficient to justify the additional cost of the extra iteration (about 26%). However, in the off-centered cases, the extra iteration produces only a minor change in stability.

#### d. Real data test

For time steps of the desired size (1800 s on a 240-km grid, 900 s on a 120-km grid), we have not yet been able to integrate the SI model version stably on the real data test case used by Skamarock et al. (2012); the model fails within a few hours, even with the inclusion of  $W$  damping or off-centering. (The model runs with time steps 10 times smaller, but this is too inefficient to be useful.) Diagnostics and sensitivity tests show that the quasi-Newton iterations fail to converge or converge very slowly, with the problem focused on the lowest model level over the steepest orography, and indicate the following explanation.

The evaluation of the horizontal pressure gradient at constant height requires a contribution from  $\partial p / \partial z$  multiplied by the slope of model levels [see (4)]. In the interior of the domain, the  $\partial p / \partial z$  contribution is

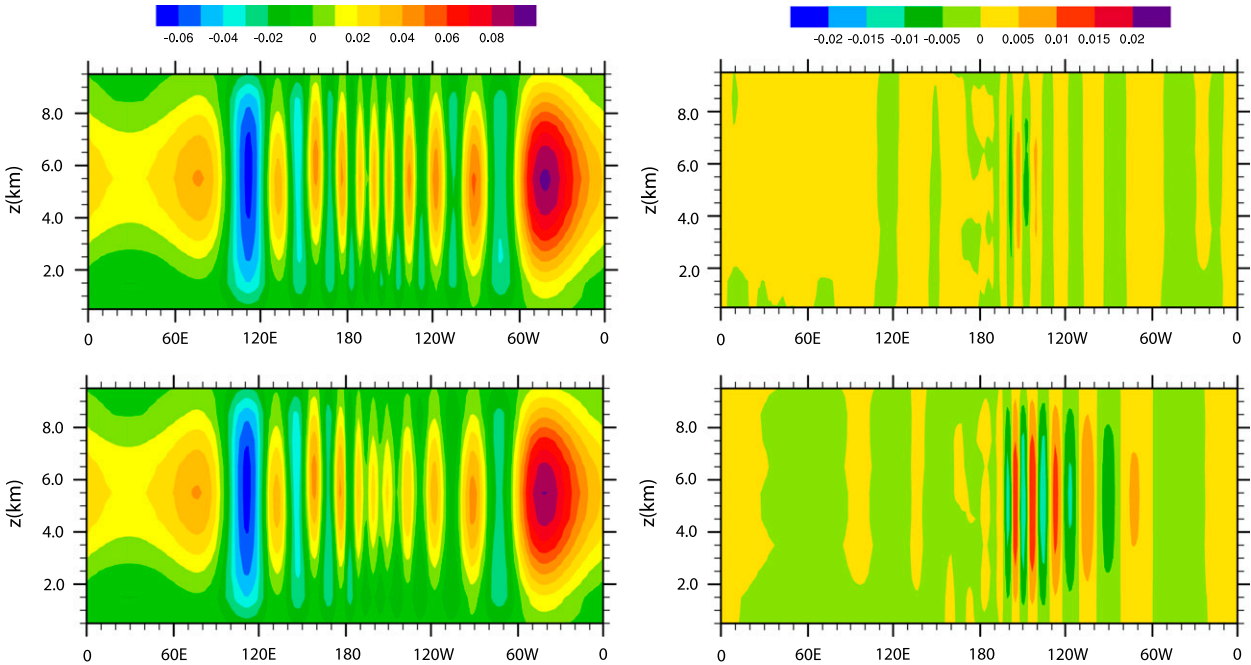


FIG. 5. Results at  $t = 3600$  s from the nonhydrostatic gravity wave test case. All panels show longitude–height sections along the equator. (top) SI time integration scheme with  $\alpha = 0.5$ ; (bottom) SI time integration scheme with  $\alpha = 0.55$ . (left) Potential temperature perturbation from the reference undisturbed profile; (right) potential temperature difference of SI minus SRK3.

interpolated vertically and horizontally to the required location. However, at the lowest model level, the  $\partial p/\partial z$  contribution is extrapolated vertically. A consequence of this extrapolation is that the sensitivity of the pressure gradient term to a pressure perturbation  $p'$  via the  $\partial p/\partial z$  contribution is comparable to or greater than the sensitivity via the  $\nabla_z p$  contribution. Scaling analysis suggests that this is likely to be the case in locations where the slope of model levels  $|z_H|$  is comparable to or greater than the ratio of vertical to horizontal grid spacing  $\Delta z/\Delta x$ . Thus, because it neglects the  $\partial p/\partial z$  term, the simplification in (26) is not a good approximation to the Jacobian of the system in such locations, and the quasi-Newton iterations do not converge well. (For comparison, the acoustic substeps of the SRK3 scheme do include the  $\partial p/\partial z$  contribution to the horizontal pressure gradient.)

We are considering two approaches that might be able to resolve this issue. The first is a reformulation of the  $\partial p/\partial z$  contribution to the horizontal pressure gradient near the bottom boundary so as to reduce the sensitivity noted above. The scheme used in END-Game (Wood et al. 2014) is one candidate. The second is to include the  $\partial p/\partial z$  contribution in the simplification in (26), and hence in the Helmholtz problem itself. This would have some cost implications: it would increase the complexity of the Helmholtz solver, and it would also increase the number of coefficients that

need to be restricted to coarser grids, though it would not affect the size of halos that need to be exchanged during the restriction, prolongation, and smoothing operations.

*e. Performance and scalability*

Model integrations of the baroclinic wave test were carried out for a range of different horizontal resolutions [from 480 km (2562 cells) to 15 km (2 621 442 cells), all with 41 levels] and using different numbers of partitioned subdomains (24–1920) in order to compare both weak and strong scalability of the SRK3 and SI model versions. Graph files were generated using the gpmets command of the Metis package (version 5.1.0) with

TABLE 2. Maximum stable time step for various model configurations.

| Scheme | No. of quasi-Newton iterations | Damping                        | Max stable $\Delta t$ (s) |
|--------|--------------------------------|--------------------------------|---------------------------|
| SRK3   | —                              | $\beta_s = 0.1, \beta_d = 0.1$ | 2970                      |
| SI     | 3                              | $\alpha = 0.5$                 | 1998                      |
| SI     | 3                              | $\alpha = 0.55$                | 4536                      |
| SI     | 3                              | $\alpha = 0.6$                 | 5130                      |
| SI     | 4                              | $\alpha = 0.5$                 | 2844                      |
| SI     | 4                              | $\alpha = 0.55$                | 4770                      |
| SI     | 4                              | $\alpha = 0.6$                 | 4644                      |

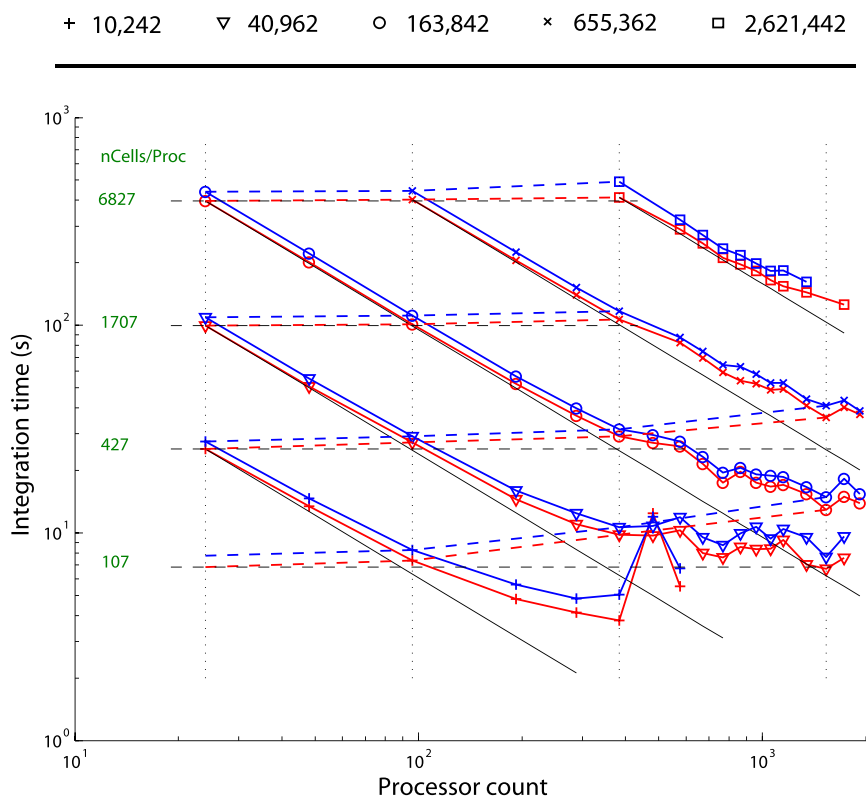


FIG. 6. Weak (dashed) and strong (solid) scaling results for the SRK3 version (red) and SI version (blue). Black reference lines indicate perfect scaling relative to a reference case with 24 processes.

default options for optimization. The scaling tests were run on the University of Exeter supercomputer Zen.<sup>3</sup> For each model version, resolution, and decomposition, the model was run once, with the code timers set to collect data from 10 consecutive periods. Each of these measured time periods represents 100 time steps, and in the results presented in this section, the minimum value from the 10 periods is used.

Figure 6 shows that the cost per time step is very similar for the SRK3 and SI model versions in all configurations, with the SI version being typically 10%–20% more expensive. In particular, both the weak and strong scaling characteristics of the two versions are very similar, with strong scaling performance falling off when there are fewer than a few hundred grid columns per

processor. Similar behavior of the SRK3 version is found on other machines.

To further understand the cost of the algorithms, timers were implemented for each code segment A–E within the main loop (including any communication within those segments) and also for all communications. Figure 7 compares the costs of the different code segments for the SRK3 and SI versions at two different resolutions and on different numbers of processors, expressed as a percentage of the total SRK3 cost. Although the behavior does not depend smoothly on processor count, some patterns are clear. The most expensive code segments are B and D, and these are significantly more expensive for the SI version, though always less than double the SRK3 cost.<sup>4</sup> Segment C is slightly cheaper for the SI version. These differences are consistent with the comparison of the algorithms in section 5. Also, as might be expected, the fractional cost

<sup>3</sup>Zen is a Silicon Graphics, Inc., (SGI) Altix Integrated Compute Environment (ICE) 8200 system. It is a water-cooled distributed-memory cluster consisting of 160 dual hex-core 2.80-GHz Intel Westmere nodes. There are 12 cores and 24 GB of memory per node, giving 1920 cores and 3.8 TB of memory in total. The compute nodes are connected with Dual DDR 4x Infiniband, and the machine uses a Linux operating system (see <http://hpc.ex.ac.uk/techspecs.html>).

<sup>4</sup>Note that, because of the way the timers were implemented, the costs for segments B and D were not cleanly separated; the total for B plus D, however, is reliable.

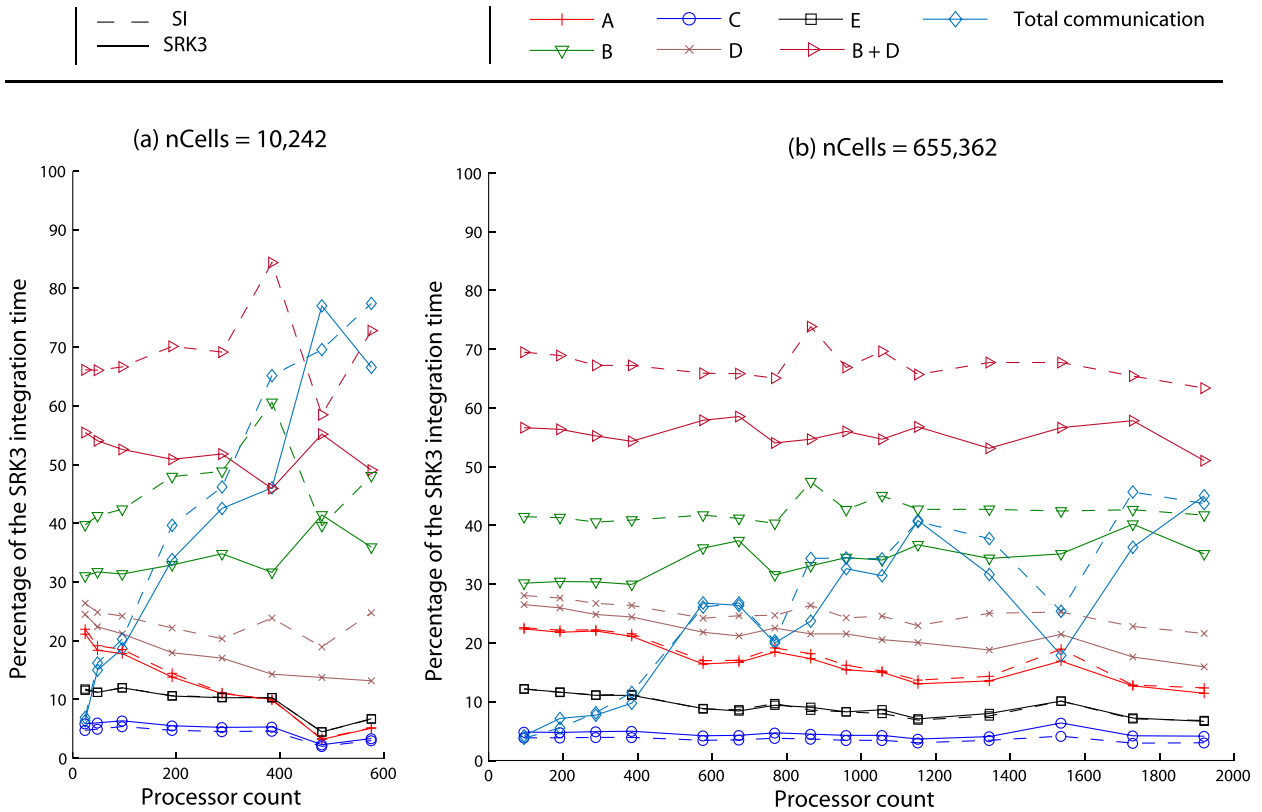


FIG. 7. Relative cost of different code segments for the SRK3 and SI versions vs number of processors, expressed as a percentage of the total SRK3 cost. Solid curves are for SRK3, and dashed curves are for SI. (a) A 240-km grid (10 242 cells); (b) a 30-km grid (655 362 cells).

of the communications gradually increases as the number of processors increases.

**7. Conclusions and discussion**

A semi-implicit formulation of the MPAS-Atmosphere dynamical core has been presented. It is based on a quasi-Newton iteration toward a Crank–Nicolson scheme. The Newton update equations lead to a Helmholtz problem similar to that in other SI models (though the unaveraging operation  $\mathcal{N}^{-1}$  that arises because of the Lorenz-grid vertical staggering does not appear to have been noticed previously). A geometric multigrid method is used to solve the Helmholtz problem.

On the Jablonowski and Williamson (2006) baroclinic wave test case and the DCMIP small-planet non-hydrostatic gravity wave test case, the SI model version produces almost identical results to the original SRK3 version, suggesting that spatial discretization errors dominate time discretization errors. The SI version costs around 10%–20% more per step than the SRK3 version. The key to achieving such efficiency in the SI version is not to do more work than necessary. Because the Helmholtz problem is embedded within the quasi-Newton iteration, it

does not need to be solved to a tight tolerance; a single V-cycle is sufficient. Moreover, the horizontal acoustic wave Courant number, which determines the horizontal length scale in the Helmholtz problem, is typically of order 10 or less; this means that a shallow V-cycle (we use three multigrid levels) is sufficient, and merging of computational subdomains is not needed. Finally, by linearizing about reference thermodynamic profiles close to the actual predicted profiles, we ensure that the quasi-Newton iteration converges quickly, and only a small number of iterations are required.

The additional cost per time step of the SI version compared to the SRK3 is compensated by the ability to take somewhat longer time steps without loss of stability. The weak and strong parallel scaling characteristics of the SI and SRK3 versions are very similar. This might be expected given the structure of the respective algorithms: both the multigrid solver in the SI version and the acoustic substepping in the SRK3 version involve a few single-layer halo exchanges per step.

We have not been able to run the SI version stably with realistic orography. Diagnostics indicate that the form of the horizontal pressure gradient term in the lowest model layer is not well captured by the approximations in the



quasi-Newton method of [section 2b](#). Further work will investigate whether an alternative form for the pressure gradient term in the lowest layer or a modification of the quasi-Newton method that makes a more complete approximation of the pressure gradient term can produce a stable method.

On locally refined spherical centroidal Voronoi grids, the relation between neighboring grids in the multigrid hierarchy becomes more complicated than in the quasi-uniform case: both the stencil and weight coefficients for the restriction and prolongation operators must be modified. We have successfully run the baroclinic wave test case using the SI model version on a locally refined grid. The details will be reported elsewhere.

Finally, we note that the code infrastructure changes implemented to handle the multigrid grid and data structures, along with the restriction and prolongation operators, may have other applications besides the SI time integration scheme; these include data assimilation and the production of quick-look, low-resolution output.

*Acknowledgments.* We are grateful to William Skamarock for valuable discussions on the MPAS formulation, and to three anonymous reviewers for their constructive comments. We also thank Rob O'Neale and David Acreman for their support in running MPAS and related codes on different computing systems. This work was funded by the U.K. Natural Environment Research Council as part of the G8 ICOMEX project under Grant NE/J005436/1.

#### REFERENCES

- Bourke, W., 1974: A multi-level spectral model. I. Formulation and hemispheric integrations. *Mon. Wea. Rev.*, **102**, 687–701, doi:[10.1175/1520-0493\(1974\)102<0687:AMLSMI>2.0.CO;2](#).
- Cullen, M. J. P., 2001: Alternative implementations of the semi-Lagrangian semi-implicit schemes in the ECMWF model. *Quart. J. Roy. Meteor. Soc.*, **127**, 2787–2802, doi:[10.1002/qj.49712757814](#).
- , and D. J. Salmond, 2003: On the use of a predictor–corrector scheme to couple the dynamics with the physical parametrizations in the ECMWF model. *Quart. J. Roy. Meteor. Soc.*, **129**, 1217–1236, doi:[10.1256/qj.02.12](#).
- Dembo, R. S., S. C. Eisenstat, and T. Steihaug, 1982: Inexact Newton methods. *SIAM J. Numer. Anal.*, **19**, 400–408, doi:[10.1137/0719025](#).
- Fulton, S. R., P. E. Ciesielski, and W. H. Schubert, 1986: Multigrid methods for elliptic problems: A review. *Mon. Wea. Rev.*, **114**, 943–959, doi:[10.1175/1520-0493\(1986\)114<0943:MMFEPA>2.0.CO;2](#).
- Heikes, R. P., D. A. Randall, and C. S. Konor, 2013: Optimized icosahedral grids: Performance of finite-difference operators and multigrid solver. *Mon. Wea. Rev.*, **141**, 4450–4469, doi:[10.1175/MWR-D-12-00236.1](#).
- Hoskins, B. J., and A. J. Simmons, 1975: A multi-layer spectral model and the semi-implicit method. *Quart. J. Roy. Meteor. Soc.*, **101**, 637–655, doi:[10.1002/qj.49710142918](#).
- Jablonowski, C., and D. L. Williamson, 2006: A baroclinic instability test case for atmospheric model dynamical cores. *Quart. J. Roy. Meteor. Soc.*, **132**, 2943–2975, doi:[10.1256/qj.06.12](#).
- Jay, L. O., 2000: Inexact simplified Newton iterations for implicit Runge–Kutta methods. *SIAM J. Numer. Anal.*, **38**, 1369–1388, doi:[10.1137/S0036142999360573](#).
- Jöckel, P., R. von Kuhlmann, M. G. Lawrence, B. Steil, C. A. M. Brenninkmeijer, P. J. Crutzen, P. J. Rasch, and B. Eaton, 2001: On a fundamental problem in implementing flux-form advection schemes for tracer transport in 3-dimensional general circulation and chemistry transport models. *Quart. J. Roy. Meteor. Soc.*, **127**, 1035–1052, doi:[10.1002/qj.49712757318](#).
- John, V., G. Matthies, and J. Rang, 2006: A comparison of time-discretization/linearization approaches for incompressible Navier–Stokes equations. *Comput. Methods Appl. Mech. Eng.*, **195**, 5995–6010, doi:[10.1016/j.cma.2005.10.007](#).
- Kar, S. J., 2006: A semi-implicit Runge–Kutta time-difference scheme for the two-dimensional shallow-water equations. *Mon. Wea. Rev.*, **134**, 2916–2926, doi:[10.1175/MWR3214.1](#).
- Klemp, J. B., W. C. Skamarock, and J. Dudhia, 2007: Conservative split-explicit time integration methods for the compressible nonhydrostatic equations. *Mon. Wea. Rev.*, **135**, 2897–2913, doi:[10.1175/MWR3440.1](#).
- , J. Dudhia, and A. D. Hassiotis, 2008: An upper gravity-wave absorbing layer for NWP applications. *Mon. Wea. Rev.*, **136**, 3987–4003, doi:[10.1175/2008MWR2596.1](#).
- Knoll, D. A., and D. E. Keyes, 2004: Jacobian-free Newton–Krylov methods: A survey of approaches and applications. *J. Comput. Phys.*, **193**, 357–397, doi:[10.1016/j.jcp.2003.08.010](#).
- Martínez, J. M., 2000: Practical quasi-Newton methods for solving nonlinear systems. *J. Comput. Appl. Math.*, **124**, 97–121, doi:[10.1016/S0377-0427\(00\)00434-9](#).
- Müller, E., and R. Scheichl, 2014: Massively parallel solvers for elliptic partial differential equations in numerical weather and climate prediction. *Quart. J. Roy. Meteor. Soc.*, **140**, 2608–2624, doi:[10.1002/qj.2327](#).
- Qaddouri, A., and V. Lee, 2011: The Canadian Global Environmental Multiscale model on the Yin–Yang grid system. *Quart. J. Roy. Meteor. Soc.*, **137**, 1913–1926, doi:[10.1002/qj.873](#).
- Robert, A., 1969: The integration of a spectral model of the atmosphere by the implicit method. *Proc. WMO/IUGG Symp. on Numerical Weather Predictions in Tokyo*, Tokyo, Japan, Japan Meteorological Agency, VII.19–VII.24.
- , J. Henderson, and C. Turnbull, 1972: An implicit time integration scheme for baroclinic models of the atmosphere. *Mon. Wea. Rev.*, **100**, 329–335, doi:[10.1175/1520-0493\(1972\)100<0329:AITISF>2.3.CO;2](#).
- Satoh, M., T. Matsuno, H. Tomita, H. Miura, T. Nasuno, and S. Iga, 2008: Nonhydrostatic icosahedral atmospheric model (NICAM) for global cloud resolving simulations. *J. Comput. Phys.*, **227**, 3486–3514, doi:[10.1016/j.jcp.2007.02.006](#).
- Skamarock, W. C., J. B. Klemp, M. G. Duda, L. D. Fowler, S.-H. Park, and T. D. Ringler, 2012: A multiscale nonhydrostatic atmospheric model using centroidal Voronoi tessellations and C-grid staggering. *Mon. Wea. Rev.*, **140**, 3090–3105, doi:[10.1175/MWR-D-11-00215.1](#).
- Ullrich, P. A., and C. Jablonowski, 2012a: MCore: A nonhydrostatic atmospheric dynamical core utilizing high-order finite-volume methods. *J. Comput. Phys.*, **231**, 5078–5108, doi:[10.1016/j.jcp.2012.04.024](#).
- , and —, 2012b: Operator-split Runge–Kutta–Rosenbrock methods for nonhydrostatic atmospheric models. *Mon. Wea. Rev.*, **140**, 1257–1284, doi:[10.1175/MWR-D-10-05073.1](#).

- , ——, J. Kent, P. H. Lauritzen, R. D. Nair, and M. A. Taylor, 2012: Dynamical Core Model Intercomparison Project (DCMIP) test case document. NCAR Tech. Doc., 83 pp. [Available online at [https://earthsystemcog.org/site\\_media/docs/DCMIP-TestCaseDocument\\_v1.7.pdf](https://earthsystemcog.org/site_media/docs/DCMIP-TestCaseDocument_v1.7.pdf).]
- Walko, R. L., and R. Avissar, 2008: The Ocean–Land–Atmosphere Model (OLAM). Part II: Formulations and tests of the non-hydrostatic dynamic core. *Mon. Wea. Rev.*, **136**, 4045–4062, doi:10.1175/2008MWR2523.1.
- Walters, D., and Coauthors, 2014: ENDGame: A new dynamical core for seamless atmospheric prediction. Met Office Tech. Rep., 26 pp. [Available online at [http://www.metoffice.gov.uk/media/pdf/s/h/ENDGameGOVSci\\_v2.0.pdf](http://www.metoffice.gov.uk/media/pdf/s/h/ENDGameGOVSci_v2.0.pdf).]
- Wicker, L. J., and W. C. Skamarock, 2002: Time-splitting methods for elastic models using forward time schemes. *Mon. Wea. Rev.*, **130**, 2088–2097, doi:10.1175/1520-0493(2002)130<2088:TSMFEM>2.0.CO;2.
- Wong, M., W. C. Skamarock, P. H. Lauritzen, and R. B. Stull, 2013: A cell-integrated semi-Lagrangian semi-implicit shallow-water model (CSLAM-SW) with conservative and consistent transport. *Mon. Wea. Rev.*, **141**, 2545–2560, doi:10.1175/MWR-D-12-00275.1.
- Wood, N., and Coauthors, 2014: An inherently mass-conserving semi-implicit semi-Lagrangian discretization of the deep-atmosphere global nonhydrostatic equations. *Quart. J. Roy. Meteor. Soc.*, **140**, 1505–1520, doi:10.1002/qj.2235.
- Yeh, K.-S., J. Côté, S. Gravel, A. Méthot, A. Patoine, M. Roch, and A. Staniforth, 2002: The CMC–MRB Global Environmental Multiscale (GEM) model. Part III: Nonhydrostatic formulation. *Mon. Wea. Rev.*, **130**, 339–356, doi:10.1175/1520-0493(2002)130<0339:TCMGEM>2.0.CO;2.
- Zängl, G., D. Reinert, P. Rípodas, and M. Baldauf, 2015: The ICON (ICOsahedral Non-hydrostatic) modelling framework of DWD and MPI-M: Description of the non-hydrostatic dynamical core. *Quart. J. Roy. Meteor. Soc.*, **141**, 563–579, doi:10.1002/qj.2378.
- Zhou, G., and S. R. Fulton, 2009: Fourier analysis of multigrid methods on hexagonal grids. *SIAM J. Sci. Comput.*, **31**, 1518–1538, doi:10.1137/070709566.