



# A compatible finite element discretisation for the nonhydrostatic vertical slice equations

C. J. Cotter<sup>1</sup>  · J. Shipton<sup>2</sup>

Received: 20 September 2022 / Accepted: 17 August 2023 / Published online: 3 September 2023  
© The Author(s) 2023

## Abstract

We present a compatible finite element discretisation for the vertical slice compressible Euler equations, at next-to-lowest order (i.e., the pressure space is bilinear discontinuous functions). The equations are numerically integrated in time using a fully implicit timestepping scheme which is solved using monolithic GMRES preconditioned by a linesmoothen. The linesmoothen only involves local operations and is thus suitable for domain decomposition in parallel. It allows for arbitrarily large timesteps but with iteration counts scaling linearly with Courant number in the limit of large Courant number. This solver approach is implemented using Firedrake, and the additive Schwarz preconditioner framework of PETSc. We demonstrate the robustness of the scheme using a standard set of testcases that may be compared with other approaches.

**Keywords** Compatible finite elements · Numerical weather prediction · Monolithic solvers · Vertical slice models

**Mathematics Subject Classification** 76U60 · 86A10 · 65M60 · 65Z05

## 1 Introduction

This article presents numerical results for a compatible finite element discretisation of the compressible Euler equations in a vertical slice geometry (i.e. two dimensional with one of the directions being the vertical). Vertical slice geometry provides an opportunity to evaluate the performance and behaviour of numerical discretisations for atmosphere dynamical cores using testcases that can be run on a laptop or workstation, forming a useful step in the development of global atmosphere dynamical cores.

---

✉ C. J. Cotter  
colin.cotter@imperial.ac.uk

<sup>1</sup> Department of Mathematics, Imperial College London, London, UK

<sup>2</sup> Mathematics and Statistics, University of Exeter, Exeter, UK

The motivation for compatible finite element methods is that they provide a naturally stable discretisation for the equations linearised about a state of rest, i.e. no numerical stabilisation (such as Riemann solvers, divergence damping, artificial viscosity, etc) is required for the wave component of the solution. This stability means that they are absent of spurious pressure modes, i.e. pressure patterns that are highly oscillatory but lead to small numerical gradients. Further, when the Coriolis force is introduced, they support exactly steady geostrophic states (Cotter and Shipton 2012), and they avoid the spurious branches of inertial oscillations (Natale et al. 2016) that are present in many finite element discretisations (such as  $P1_{DG}$ -P2 and CR1-P0 on triangles). This satisfies the main requirements set out in (Staniforth and Thuburn 2012), which became a wishlist for discretisations considered in the Gung Ho project designing a new atmosphere dynamical core for the Met Office. Natale et al. (2016) also showed that if an appropriate space is chosen for potential temperature as proposed by Guerra and Ullrich (2016) (guided by the Lorenz staggering for finite difference methods), there are no spurious hydrostatic modes. Compatible finite element methods are also flexible in allowing to choose finite element spaces so that there are sufficient velocity degrees of freedom per pressure degree of freedom (avoiding the spurious inertia-gravity wave modes present in triangular C-grid discretisations (Danilov 2010)). Finally, they are consistent on very general meshes (arbitrary triangulations with some minimum aspect ratio, and quadrilateral meshes coming from piecewise smooth maps applied to regular grids) and spaces can be selected of arbitrary high order consistency. In particular, the Coriolis term is consistent on cubed sphere meshes, avoiding an issue discovered with the C-grid discretisation when applied to cubed sphere or dual icosahedral meshes when formulated to satisfy the properties above (Thuburn and Cotter 2012; Thuburn et al. 2014). The numerical weather prediction community is moving towards such grids because they avoid the parallel bottlenecks associated with the poles in the latitude longitude grids in current and previous use. It was for this reason that compatible finite element methods were selected for the Gung Ho dynamical core, which is built around the lowest order spaces currently (Sergeev et al. 2023; Melvin et al. 2019).

Compatible finite element methods also lend themselves to variational and conservative formulations. These formulations are the result of extension of similar schemes constructed using the C-grid finite difference staggering (Arakawa and Lamb 1981; Gassmann 2013; Dubos et al. 2015). Taylor et al. (2020) have recently provided a formulation using spectral element methods.

McRae and Cotter (2014) provided a scheme for the rotating shallow water equations that conserves energy and enstrophy. This has been extended to bounded domains (Bauer and Cotter 2018), upwinded formulations that dissipate enstrophy but preserve energy (Wimmer et al. 2020), and vertical slice model (Wimmer et al. 2021). An alternative set of compatible finite element spaces based on splines was presented in Eldred et al. (2019), and a related approach based on mimetic spectral elements was presented in Lee and Palha (2018, 2020).

A variational discretisation (i.e., a discretisation derived from Hamilton's principle) for the two dimensional incompressible Euler equations was derived in Natale and Cotter (2018), which was further developed for more general fluid models in Gawlik and Gay-Balmaz (2020, 2022). We are not using variational or conservative formulations in this paper, but we note that our formulation is rather close to them, which

might be expected to reduce spurious energy transfers between kinetic, potential and internal energy.

The lowest order compatible finite element spaces (i.e., those spaces with RT0 for velocity and P0 used for pressure/density) are closely related to the C grid finite difference method popular amongst many operational weather models (Wood et al. 2014, for example). However, these spaces are only first order accurate, so a finite volume approach is required where one is in effect solving for cell averaged quantities, and higher order accuracy for those cell averaged quantities must be obtained by using finite volume transport schemes, which have a stencil over several cells. This was done in (Melvin et al. 2019). The advantage of using the next-to-lowest-order (NLO) spaces (RT1 is used for velocity, and P1<sub>DG</sub> used for pressure/density) is that they are naturally second order accurate, so standard finite element formulations are sufficient. This leads to a very clean formulation where assembly only requires to fetch data from a single cell or from pairs of cells joined by facets. This makes it particularly amenable to automated code generation of MPI parallel codes using tools such as Firedrake (Rathgeber et al. 2016). (Shipton et al. 2018) presented a practical scheme for the rotating shallow water equations using NLO spaces. In this paper we present a practical scheme for the compressible Euler equations using NLO spaces, evaluated using a standard suite of vertical slice model testcases. A related scheme was coupled with a moisture model in Bendall et al. (2020), but was not benchmarked against the tests considered here. We use a fully implicit timestepping method, solved using a monolithic iterative solver i.e. the full coupled system of all variables is solved together without elimination. The equations are solved using GMRES, using a linesmoothener which acts as an additive Schwarz method with each patch comprising the “star” of a vertical edge, i.e. all degrees of freedom associated with the interior of the vertical column surrounding that edge. The columnar approach is necessary due to the thin domain geometry occurring in atmosphere models. Numerical analysis of these schemes is difficult, but they are motivated by finding monolithic smootheners that result in a (block) Jacobi iteration for the density/pressure variable after elimination within the patch. For example, a similar method (in local rather than columnar form) was proposed in Adler et al. (2021); Laakmann et al. (2022).

There are a few reasons for including this approach here. First, it is useful to focus on the behaviour of the spatial discretisation without clouding the issue with aspects related to the splitting of advection and wave propagation, for example. In a monolithic approach, the basic code becomes very simple, and the challenges are exported to the iterative solver, which is provided here by the PETSc library. Second, the monolithic approach is itself new, and the paper provides a useful demonstration that it is useable in practice for numerical weather prediction.

The rest of the article is structured as follows. Section 2 presents the discretisation in space and time, together with the iterative solution strategy for the resulting implicit system of equations, and our strategy for obtaining hydrostatic balance reference profiles. Section 3 presents the numerical examples, and Sect. 4 provides a summary and outlook.

## 2 Discretisation

### 2.1 Governing equations

Our description of the two-dimensional nonhydrostatic dry Euler equations can remain quite brief because our approach discretises the equations in Cartesian coordinates (even on terrain following coordinates or in spherical geometry). We write the equations in  $\theta - \Pi$  (potential temperature-Exner pressure) form,

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} + f \hat{\mathbf{k}} \times \mathbf{u} + c_p \theta \nabla \Pi + g \hat{\mathbf{k}} = 0, \quad (1)$$

$$\frac{\partial \theta}{\partial t} + \mathbf{u} \cdot \nabla \theta = 0, \quad (2)$$

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\mathbf{u} \rho) = 0, \quad (3)$$

$$\Pi^{(1-\kappa)/\kappa} = \frac{R}{p_0} \rho \theta, \quad (4)$$

where  $\mathbf{u}$  is the velocity,  $\theta$  is the potential temperature,  $\rho$  is the density,  $\Pi$  is the Exner pressure,  $f$  is the Coriolis parameter,  $\hat{\mathbf{k}}$  is the unit vector pointing upwards,  $c_p$  is the specific heat at constant pressure,  $R$  is the gas constant,  $p_0$  is the reference pressure,  $g$  is the acceleration due to gravity and  $\kappa = R/c_p$ . Here, we have presented the equations as they would be read in three-dimensional form. In the vertical slice model, one restricts the spatial dependence of the fields to the  $x - z$  plane, whilst retaining three Cartesian components of the vector field, i.e.  $\mathbf{u}(\mathbf{x}, t) : \mathbb{R}^2 \times [0, T] \rightarrow \mathbb{R}^3$ . In a code, a simple way to implement this is to use a three dimensional mesh that is one cell wide in the  $y$ -direction, and periodic in that boundary condition. If a higher order finite element is used, it is still possible to represent variations in the  $y$ -direction. We expect  $y$ -independent solutions should be preserved by symmetry, but care should be taken to check that this is the case. In fact this is an advantageous approach, since the same code can be used for vertical slice and full 3D models, and validation in the vertical slice case adds to confidence in the full 3D model.

In this paper, we use the vector-invariant form of the advection term,

$$(\mathbf{u} \cdot \nabla) \mathbf{u} = (\nabla \times \mathbf{u}) \times \mathbf{u} + \frac{1}{2} |\nabla \mathbf{u}|^2. \quad (5)$$

Some of the test problems we will consider contain a Newtonian damping term to absorb radiating internal waves as they reach the upper boundary, requiring the addition of the following term,

$$\mu \hat{\mathbf{k}} \mathbf{u} \cdot \hat{\mathbf{k}}, \quad (6)$$

to the left hand side of Eq. (1) where  $\mu$  is the (spatially-dependent) damping parameter specific to the test problem (which we will specify later), and is otherwise zero.

Similarly, one of the test problems requires the addition of viscous and diffusion terms in order to observe convergence with mesh resolution, in which case the same diffusion parameter  $\nu$  is used. Then we add

$$- \nu \nabla^2 \mathbf{u} \quad (7)$$

to the left hand side of Eq. (1), and

$$- \nu \nabla^2 \theta \quad (8)$$

to the left hand side of Eq. (2). These terms are not necessary for stability and are only included to facilitate convergence testing and comparison with other published results.

We call the computational domain  $\Omega$ , which in this paper is always a rectangle with lateral periodic boundary conditions (or a deformation of a rectangle to accommodate topography), and slip boundary conditions  $\mathbf{u} \cdot \mathbf{n} = 0$  on the bottom and sides, where  $\mathbf{n}$  is the unit normal to the boundary. When diffusion and viscosity are included, this boundary condition is augmented by  $\frac{\partial \theta}{\partial n} = 0$  and  $\frac{\partial \mathbf{u} \times \mathbf{n}}{\partial n} = 0$ .

## 2.2 Spatial discretisation

### 2.2.1 Meshes and finite element spaces

In this paper, we make use of two types of meshes. For problems with  $f = 0$  and no out-of-plane component to the velocity, we construct structured meshes of regular rectangles. For problems with  $f \neq 0$ , we construct structured meshes of regular cuboids that are one cell wide and periodic in the  $y$ -direction (to facilitate efficient solution of  $y$ -independent problems and to allow seamless transition to fully 3D problems). In either case, for problems with topography we apply a terrain following transformation to the mesh of the form  $z \mapsto z + \phi(x, y, z)$ , which preserves column structure (vertical faces of cells remain vertical) but does deform rectangles into trapezia and cuboids into trapezium prisms.

Following McRae et al. (2016), we select the finite element spaces as follows. The velocity space  $\mathbb{V}_1$  is the Raviart-Thomas space of degree 1 (RT1) on hexahedra (see below for more discussion about the vertical slice case), the density space  $\mathbb{V}_2$  is the discontinuous trilinear space (bilinear in 2D), and temperature space  $\mathbb{V}_\theta$  is the tensor product of quadratic functions in the vertical, and bilinear functions (linear in 2D) in the horizontal. It is continuous in the vertical, and discontinuous in the horizontal. This choice mirrors the structure of the vertical component of the velocity space, to facilitate the representation of hydrostatic balance (see Natale et al. (2016) for an analysis of this, and a more detailed description of these spaces). See Bercea et al. (2016) for information about how finite element methods with these spaces can be used performantly.

In 3D, the  $\mathbb{V}_1$  space uses a biquadratic construction: the vertical component of velocity in the reference cube is quadratic in the vertical direction, and linear in the

horizontal directions, with a symmetric pattern applied to the other component(s). If an  $x - z$  planar mesh is used in a vertical slice model, then the  $x - z$  components of velocity are represented in the RT1 space on quadrilaterals, which the  $y$  component is discontinuous and bilinear (same as the density). The RT1 functions are mapped into physical cells using the Piola mapping (Rognes et al. 2010). This guarantees continuity of normal components of the functions across cell facets at the expense of replacing polynomials by rational polynomials when the cells are trapezia or trapezium prisms.

Since these finite element spaces all contain the complete space of linear polynomials, classical approximation theory indicates that they can approximate smooth functions on the reference element with second order error in the  $L^2$  norm. There is a technicality that the Piola transformation means that  $\mathbb{V}_1$  on mesh elements with terrain following meshes do not contain all linear polynomials. However, Natale et al. (2016) showed that second order approximation is still obtained if the mapping from a rectangular domain is smooth (or, piecewise smooth provided that the number of pieces is fixed).

The spatial discretisation of the Eqs. (1–4) is then to find  $(\mathbf{u}(t), \rho(t), \theta(t)) \in \hat{\mathbb{V}}_1 \times \mathbb{V}_2 \times \mathbb{V}_\theta$  such that

$$\begin{aligned} & \int_{\Omega} \mathbf{w} \cdot \mathbf{u}_t + \mu \mathbf{w} \cdot \hat{\mathbf{k}} \mathbf{u} \cdot \hat{\mathbf{k}} + f \mathbf{w} \cdot \hat{\mathbf{k}} \times \mathbf{u} \, dx + \mathbf{w} \cdot \hat{\mathbf{k}} g \, dx \\ & + \underbrace{\int_{\Omega} \nabla_h \times (\mathbf{w} \times \mathbf{u}) \times \mathbf{u} \, dx - \int_{\Omega} \nabla \cdot \mathbf{w} \frac{1}{2} |\mathbf{u}|^2 \, dx + \int_{\Gamma} \llbracket \mathbf{n} \times (\mathbf{u} \times \mathbf{w}) \rrbracket \cdot \tilde{\mathbf{u}} \, dS}_{\text{from } (u \cdot \nabla) u} \\ & - \underbrace{\int_{\Omega} \nabla_h \cdot (\mathbf{w} \theta) c_p \Pi \, dx + \int_{\Gamma_v} \llbracket \mathbf{n} \cdot \mathbf{w} \theta \rrbracket c_p \{ \Pi \} \, dS}_{\text{from } c_p \theta \nabla \Pi} = 0, \quad \forall \mathbf{w} \in \hat{\mathbb{V}}_1, \end{aligned} \tag{9}$$

$$\begin{aligned} & \int_{\Omega} q \theta_t - \nabla_h \cdot (\mathbf{u} q) \theta \, dx + \underbrace{\int_{\Gamma_v} \llbracket q \mathbf{n} \cdot \mathbf{u} \rrbracket \tilde{\theta} \, dS}_{\text{horizontal upwinding}} \\ & + \underbrace{\int_{\Gamma} C_0 h^2 |\mathbf{u} \cdot \mathbf{n}| \llbracket \nabla_h q \rrbracket \cdot \llbracket \nabla_h \theta \rrbracket \, dS}_{\text{edge stabilisation}} = 0, \quad \forall q \in \mathbb{V}_\theta, \end{aligned} \tag{10}$$

$$\int_{\Omega} \phi \rho_t - \nabla_h \phi \cdot \mathbf{u} \rho \, dx + \underbrace{\int_{\Gamma} \llbracket \phi \mathbf{u} \cdot \mathbf{n} \rrbracket \tilde{\rho} \, dS}_{\text{upwinding}} = 0, \quad \forall \phi \in \mathbb{V}_2, \tag{11}$$

where

1.  $\nabla_h$  indicates the “broken” gradient obtained by evaluating the gradient separately in each cell,
2.  $\Gamma$  is the set of interior facets (with  $\Gamma_v$  the set of vertical facets between adjacent columns),
3.  $\llbracket \psi \rrbracket$  is the “jump” operator applied to a quantity  $\psi$  returning  $\psi^+ - \psi^-$  where each interior facet has sides arbitrarily labelled “+” and “-” (and noting that  $\mathbf{n}$  is the unit normal to a facet with  $\mathbf{n}^+$  pointing into the + side and *vice versa* for  $\mathbf{n}^-$ ),

4.  $\tilde{\psi}$  indicates the upwind value of any quantity  $\psi$  on the facet i.e. the value on the side where  $\mathbf{u} \cdot \mathbf{n} \geq 0$  (making an arbitrary choice not affecting the result when  $\mathbf{u} \cdot \mathbf{n} = 0$ ),
5.  $\{\psi\}$  is the average operator returning  $(\psi^+ + \psi^-)/2$  when evaluated on a facet,
6.  $\Pi$  is defined according to (4) but now applied to the numerical approximations  $\rho$  and  $\theta$  (so that we do not separately solve for an independent variable  $\Pi$ ),
7.  $h$  is an estimate of the cross facet meshscale defined by  $\{V_e\}/A_f$  with  $V_e$  being the (cellwise constant) cell volume, and  $A_f$  being the (facetwise constant) facet area,
8.  $C_0$  is an edge stabilisation constant (chosen here to be  $2^{-7/2}$  following the suggested scaling  $p^{-7/2}$  with polynomial degree  $p$  given by Burman and Ern (2007)), and
9.  $\mathring{\mathbb{V}}_1$  is the subspace of  $\mathbb{V}_1$  containing all functions that satisfy the  $\mathbf{u} \cdot \mathbf{n} = 0$  boundary condition on the top and bottom of the domain.

The treatment of the velocity advection term was first demonstrated (for an incompressible Boussinesq vertical slice model) by Yamazaki et al. (2017), inspired by Natale and Cotter (2018) but different from the energy conserving form for compressible Euler equations proposed in Wimmer et al. (2021). The surface term deals with the fact that the velocity space does not have continuous tangential components in general; it also has a stabilising effect as examined in Natale and Cotter (2017). The treatment of the pressure gradient term first appeared in Natale et al. (2016), and has been used in a modified energy conserving form in Wimmer et al. (2021), and with lowest order spaces in Bendall et al. (2020) and as part of the Gung Ho dynamical core formulation in Melvin et al. (2019). The treatment of the potential temperature advection term deviates here from the ‘‘SUPG’’ formulation proposed in Yamazaki et al. (2017), using an edge stabilisation proposed by Burman (2005) instead, in combination with standard discontinuous Galerkin style upwinding on vertical faces. Even though the temperature space is only continuous in the vertical direction, we found that edge stabilisation was necessary on all faces to achieve stable results.

It should be noted that in the case of spatially varying topography, it is not possible to compute integrals exactly because of the  $\det(J)$  in the reciprocal appearing in the velocity basis functions (due to the Piola mapping), where  $J$  is the Jacobian of the reference element to mesh element mapping. Thus, we have to approximate the integrals using numerical quadrature, and we have to select a quadrature degree for each term so that the discretisation is stable and sufficiently accurate. Here we just take the suggested quadrature degree from the heuristic computed in UFL (Alnæs et al. 2014) which selects a (rather generous) quadrature rule for approximate integration, which is certainly sufficient for stability and second order consistency. In fact, many of the integrals are nevertheless computed exactly, due to fortuitous cancellations that are discussed in (Cotter and Thuburn 2014). Another source of inexact quadrature is the function  $\Pi$  which involves a noninteger power of  $\rho$  and  $\theta$ .

When test cases require a viscous term, we use the symmetric interior penalty discretisation (to deal with discontinuities in the tangential component of the velocity) as described in (Cockburn et al. 2007), in which case the term

$$\nu \left( \int_{\Omega} \nabla_h \mathbf{u} : \nabla_h \mathbf{v} \, dx - \int_{\Gamma} \llbracket \mathbf{v} \otimes \mathbf{n} \rrbracket : \{\nabla \mathbf{u}\} \, dS - \int_{\Gamma} \llbracket \mathbf{u} \otimes \mathbf{n} \rrbracket : \{\nabla \mathbf{v}\} \, dS \right)$$

$$+ \int_{\Gamma} \frac{\eta}{h} [\mathbf{u} \otimes \mathbf{n}] : [\mathbf{v} \otimes \mathbf{n}] \, dS \quad (12)$$

is added to the left hand side of Eq. (9), where  $\nu$  is the dynamic viscosity,  $:$  indicates the double contraction for tensors (so that  $A : B = \sum_{ij} A_{ij} B_{ij}$ ), and  $\eta$  is a dimensionless penalty parameter, chosen here to have the value 10 which is experimentally determined to produce a stable discretisation for both  $\nabla_1$  and  $\nabla_\theta$ . A similar formula (with the same parameters, but adapted to scalar fields) is used when test cases require a diffusion term in the potential temperature equation.

### 2.3 Time discretisation and iterative solver

The time discretisation used is the implicit midpoint rule, a fully implicit second order method. This is obtained by replacing time derivatives by time differences e.g.  $\theta_t \mapsto (\theta^{n+1} - \theta^n)/\Delta t$ , and replacing all other occurrences of fields by their midpoint values e.g.  $\theta \mapsto (\theta^{n+1} + \theta^n)/2$ . This nonlinear system is then solved for the  $n + 1$  variables using the linesearch Newton method provided by PETSc (Balay et al. 2020) using Firedrake's NonlinearVariationalSolver object. The resulting Jacobian systems are solved using GMRES applied to the full monolithic velocity-density-temperature system, preconditioned by an additive Schwarz method, which does direct solves over column patches surrounding one vertex on the base mesh (in vertical slice models this corresponds to forming a patch out of two neighbouring cells). In fact, a “star” patch is used, which neglects degrees of freedom associated with the horizontal boundaries of the patch. This patch is formed out of the “star iteration” (see Sect. 4 in (Farrell et al. 2019)) applied to a vertex in the base mesh, before extruding up the column. These direct solves couple all three fields, and the direct solve uses PETSc's own LU factorisation algorithm using a reverse Cuthill-McGee ordering to reduce the fill in. This algorithm can be and is parallelised using domain decomposition, which was automated in our implementation using Firedrake. In all of our test cases, Newton's method converges in 2–3 iterations, and GMRES converges in 10–35 iterations depending on the flow complexity, independent of mesh size provided that a constant Courant number is maintained whilst decreasing the mesh size.

This solver approach is necessary because we chose to use a fully implicit timestepping scheme, solved using Newton's method. This requires us to be able to solve systems  $Jx = b$  where  $J$  is the full Jacobian obtained by linearisation about the state at each Newton iteration. In the case where  $J$  is the linearisation about a state of rest,  $J$  can be reduced to a positive definite Schur complement using the hybridisation technique provided the discretisation proposed in Betteridge et al. (2022) is used. However, this technique does not work when the linearisation is calculated for states with nonzero velocities, since these introduce additional intercell coupling.

### 2.4 Hydrostatic balance

The test cases that we consider here require the computation of a background density profile  $\rho_b$  that is in hydrostatic balance given the specified potential temperature  $\theta_b$ .



We use a nonhydrostatic model, but testcases require an initialisation at a state of hydrostatic balance, which we describe here. To avoid unphysical motion at lower resolutions we compute this hydrostatic balance numerically, i.e. we require that

$$\begin{aligned}
 & - \int_{\Omega} \nabla_h \cdot (\mathbf{w}\theta_b)_{c_p} \Pi_b \, dx + \int_{\Gamma} \llbracket \mathbf{n} \cdot \mathbf{w}\theta_b \rrbracket_{c_p} \{ \Pi_b \} \, dS \\
 & + \int_{\Omega} \mathbf{g}\mathbf{w} \cdot \hat{\mathbf{k}} \, dx = 0, \quad \forall \mathbf{w} \in \mathring{\mathbb{V}}_{1,v},
 \end{aligned} \tag{13}$$

where  $\mathbb{V}_{1,v}$  the vertical subspace of  $\mathbb{V}_1$ , and  $\mathring{\mathbb{V}}_{1,v}$  is the corresponding vertical subspace of  $\mathring{\mathbb{V}}_1$ .<sup>1</sup> We note that  $\Pi_b$  is just a local nonlinear function of  $\rho_b$  and  $\theta_b$ . As discussed in Natale et al. (2016), we can solve this equation for  $\rho_b$  by introducing an auxiliary variable  $\mathbf{v}$  (which will turn out to vanish), and solving the coupled system

$$\begin{aligned}
 & \int_{\Omega} \mathbf{w} \cdot \mathbf{v} \, dx - \int_{\Omega} \nabla_h \cdot (\mathbf{w}\theta_b)_{c_p} \Pi_b \, dx \\
 & + \int_{\Omega} \mathbf{g}\mathbf{w} \cdot \hat{\mathbf{k}} \, dx + \int_{\partial\Omega_0} c_p \mathbf{w} \cdot \mathbf{n}\theta_b \Pi_0 \, dS = 0, \quad \forall \mathbf{w} \in \tilde{\mathbb{V}}_{1,v},
 \end{aligned} \tag{14}$$

$$\int_{\Omega} \nabla_h \cdot (\mathbf{v}\theta_b)_{c_p} \phi \, dx = 0, \quad \forall \phi \in \mathbb{V}_2, \tag{15}$$

where  $\partial\Omega_0$  is the surface where we have chosen to set  $\Pi_b = \Pi_0$  as a boundary condition (the bottom boundary for the test cases in this paper),  $\tilde{\mathbb{V}}_{1,v}$  is the subspace of  $\mathbb{V}_{1,v}$  containing functions that satisfy  $\mathbf{w} \cdot \mathbf{n} = 0$  on the opposite boundary (the top boundary for the test cases in this paper). We note that the surface integral in Eq. (13) vanishes when  $\mathbf{w} \in \mathbb{V}_{1,v}$  because  $\mathbf{w} \cdot \mathbf{n} = 0$  on vertical faces, and we note that the weak boundary condition integral in (14) over  $\partial\Omega_0$  vanishes when  $\mathbf{w} \in \mathring{\mathbb{V}}_{1,v} \subset \tilde{\mathbb{V}}_{1,v}$ . Since  $\mathbf{v}$  vanishes at the solution, we recover the hydrostatic condition (13). The system (14–15) decouples into independent columns, which we solve using Newton’s method. Natale et al. (2016) showed that the linearisation around a given state of this system is well-posed, and we solve the resulting linear systems for  $(\rho_b, \mathbf{v})$  updates directly. We find an initial guess for  $\rho_b$  by first solving the corresponding linear system for  $(\Pi_b, \mathbf{v})$  where  $\Pi_b \in \mathbb{V}_2$  is taken as an independent variable instead of a local function of  $\rho_b$  and  $\theta_b$ . We then project the formula for  $\rho_b$  in terms of  $\theta_b$  and  $\Pi_b$  into  $\mathbb{V}_2$ . If we use the result as an initial guess then Newton’s method converges in 2–3 iterations. Note that this computation is only done to compute a set of initial conditions, and is not used in the forward model which is always nonhydrostatic.

### 3 Computational examples

In this section we demonstrate our discretisation approach using the suite of test problems considered in Melvin et al. (2010). This suite tests the vertical slice discretisation

<sup>1</sup> In the presence of orography,  $\mathbb{V}_{1,v}$  does not contain all of the vertical component of the velocity, and solving this equation will lead to pressure gradient errors. This is described further in Natale et al. (2016).

on a range of flows that are relevant to numerical weather prediction, including acoustic and gravity waves and flows driven by both buoyancy and orography. The gravity wave and orographic wave tests are run in both the hydrostatic and nonhydrostatic regimes. To avoid the need to refer back, we provide a brief summary of the test problems here. Some constants that are consistent across all tests are provided in Table 1. In each test we have used the same timestep  $\Delta t$  values as Melvin et al. (2010), and double the values of  $\Delta x$  and  $\Delta z$ , which ensures the same number of degrees of freedom (because we are using NLO spaces). A summary of resolutions, timesteps, number of iterations, and the timings is provided in Table 2.

### 3.1 Gravity waves

This test case is the ‘‘Hello World!’’ of vertical slice test problems, first proposed in Skamarock and Klemp (1994). There are two versions of the test case, the nonhydrostatic flow regime version with velocity constrained to the  $x-z$  plane and consequently  $f = 0$ , and the hydrostatic flow regime version with 3D velocity and  $f = 10^{-4} \text{ s}^{-1}$ . Note that we are solving the nonhydrostatic equations in both versions, and the naming just describes the asymptotic flow regime and not the equations solved. The domain is given by  $L/2 \leq x \leq L/2$  and  $0 \leq z \leq H$  where  $L = 3 \times 10^5 \text{ m}$  in the nonhydrostatic case and  $L = 6 \times 10^6 \text{ m}$  in the hydrostatic case. In both cases the height is  $H = 10^4 \text{ m}$  and there are periodic boundary conditions in the horizontal direction.

In both cases, the potential temperature is initialised to a background profile

$$\theta_b = T_{\text{surf}} \exp(N^2 z/g), \quad T_{\text{surf}} = 300 \text{ K}, \quad (16)$$

before solving for the hydrostatically balanced  $\rho_b$ . Then a perturbation is added to the potential temperature,

$$\Delta\theta = \Delta\theta_0 \frac{\sin(\pi z/H)}{1 + x^2/a^2}, \quad (17)$$

where  $\Delta\theta_0 = 10^{-2} \text{ K}$  and  $a = 5 \times 10^3 \text{ m}$  for the nonhydrostatic flow regime and  $a = 10^5 \text{ m}$  for the hydrostatic flow regime. In both cases, the horizontal velocity in the  $x$ -direction is initialised to  $20 \text{ ms}^{-1}$  and the other components to zero. In the hydrostatic case, an additional forcing term is introduced to balance the Coriolis force, adding  $f \times (0, -20, 0)$  to the left hand side of Eq. (1).

Plots of the nonhydrostatic and hydrostatic flow regime solutions are shown in Figs. 1 and 2, respectively. These solutions closely match the results from Melvin et al. (2010) at similar resolutions.

### 3.2 Density current

This test is taken from the classic intercomparison project of Straka et al. (1993), simulating a dense bubble in an isentropic, hydrostatic atmosphere. The domain is

**Table 1** Some constants that take consistent values across all the computational examples

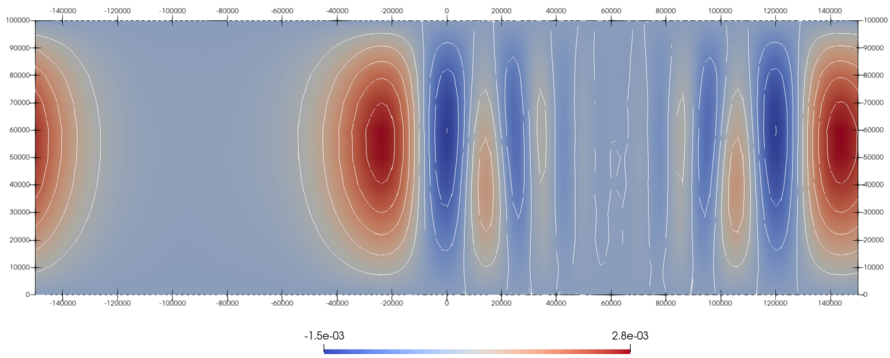
$g$	$N$	$f$	$c_p$	$R$	$p_0$	$c_v$
9.810616 ms <sup>-2</sup>	10 <sup>-2</sup> s <sup>-1</sup>	10 <sup>-4</sup> s <sup>-1</sup>	1004.5 Jkg <sup>-1</sup> K <sup>-1</sup>	287 Jkg <sup>-1</sup> K <sup>-1</sup>	1000 hPa	717 Jkg <sup>-1</sup> K <sup>-1</sup>

The exceptions are the nonhydrostatic variants of the gravity wave and orographic flows and the density current which are non-rotating so  $f = 0$ ; the density current also has an isentropic background state so  $N = 0$

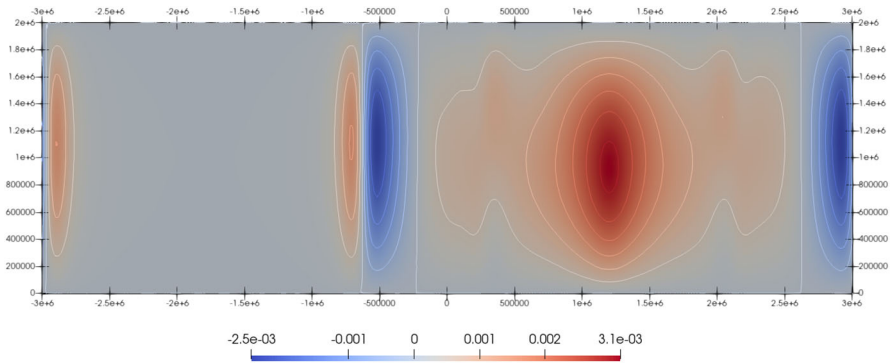
**Table 2** Table showing the resolution,  $\Delta t$ , number of GMRES iterations per timestep (over all the Newton iterations), the time taken and the number of cores used, for each of the testcases.

Testcase	Columns	Layers	$\Delta t$ (s)	GMRES its per step	Time (s)	Cores
Nonhydrostatic gravity wave	150	5	12	31.512	48.789	16
Hydrostatic gravity wave	300	10	100	31.022	227.51	16
Nonhydrostatic mountain	180	70	5	27.239	1912.5	16
Hydrostatic mountain	100	60	20	33.959	6521.6	16
Density current $\Delta x = 800$ m	64	8	4	62.8	60.04	16
Density current $\Delta x = 400$ m	128	16	2	55.05	208.4	16
Density current $\Delta x = 200$ m	256	32	1	49.11	1157	16
Density current $\Delta x = 100$ m	512	64	0.5	43.46	8452	32
Schär mountain	100	50	8	49.802	1699.1	16
Schär mountain	100	50	40	274.756	1342.6	16

The timings include the nonhydrostatic initialisation for the density and solver setups



**Fig. 1** Contour plot of vertical velocity for the nonhydrostatic gravity wave at time  $t = 3000$  s at resolution  $\Delta x = 2000$  m,  $\Delta z = 2000$  m, and  $\Delta t = 12$  s. Contours are drawn every  $5 \times 10^{-4} \text{ms}^{-1}$



**Fig. 2** Contour plot of vertical velocity for the hydrostatic gravity wave at time  $t = 60,000$  s at resolution  $\Delta x = 20,000$  m,  $\Delta z = 1000$  m, and  $\Delta t = 100$  s. Contours are drawn every  $5 \times 10^{-4} \text{ms}^{-1}$

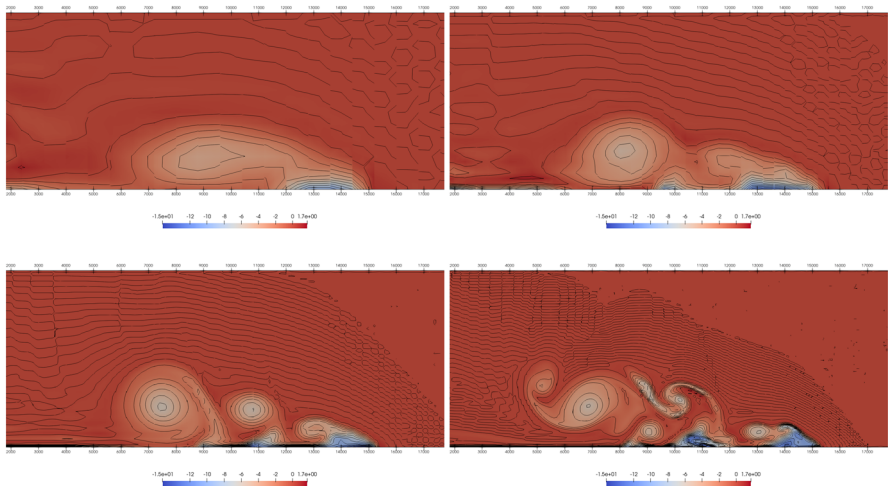
$-L/2 \leq x \leq L/2$  where  $L = 51200$  m, and  $0 \leq z \leq H = 6400$  m with periodic boundary conditions in the horizontal direction.

The background temperature profile is chosen to be isentropic, i.e. the potential temperature is constant. In this case the background potential temperature is  $\theta_b = 300\text{K}$ . The background density profile is then obtained by solving for hydrostatic balance for this potential temperature profile, with the boundary condition  $\Pi = 1$  on the bottom boundary. We then apply a perturbation to the temperature,

$$\Delta T = \begin{cases} 0 & \text{if } L_r > 1, \\ -15 (\cos(\pi L_r) + 1) / 2 & \text{otherwise,} \end{cases} \tag{18}$$

at constant pressure  $p$ , where

$$L_r = \sqrt{\left(\frac{x-x_c}{x_r}\right)^2 + \left(\frac{z-z_c}{z_r}\right)^2}, \tag{19}$$



**Fig. 3** For the density current testcase: contours of  $\Delta\theta$  at 15 min. Contour intervals are  $1K$ , with the first contour being at  $-1 K$ . From left to right and top to bottom, the resolutions are  $(\Delta x, \Delta t) = (800 \text{ m}, 4 \text{ s}), (400 \text{ m}, 2 \text{ s}), (200 \text{ m}, 1 \text{ s})$  and  $(100 \text{ m}, 0.5 \text{ s})$  respectively

and  $(x_c, x_r) = (0 \text{ m}, 4000 \text{ m}), (z_c, z_r) = (3000 \text{ m}, 2000 \text{ m})$ . This corresponds to making the potential temperature perturbation  $\Delta\theta = \Delta T / \Pi$ , where  $\Pi$  is the background Exner pressure profile, and then perturbing the density according to  $(\rho + \Delta\rho)(\theta + \Delta\theta) = \rho\theta$ . In our implementation,  $\Delta\theta$  was computed by the following steps:

1. Project the formula  $\Delta T / \Pi(\rho_b, \theta_b)$  into  $\mathbb{V}_\theta$ , where  $\rho_b \in \mathbb{V}_2$  and  $\theta_b \in \mathbb{V}_\theta$  are the previously computed initial conditions for density and temperature respectively. This is the initial condition  $\theta_0$  for potential temperature.
2. Project the formula  $\rho_b\theta_0/\theta_b$  into  $\mathbb{V}_2$ . This is the initial condition  $\rho_0$  for density.

These projections are  $L^2$  projections using incomplete but high order quadrature for the integrals over the formulae, similar to those used in the dynamical equations.

Without viscosity, this problem becomes ill-posed after forming a singular vorticity structure in finite time. Hence, to be able to compare between models, a viscosity of  $\nu = 75 \text{ m}^2 \text{ s}^{-1}$  is included in the velocity equation and a diffusivity of  $\kappa = 75 \text{ m}^2 \text{ s}^{-1}$  is included in the potential temperature equation.

For this testcase, the comparison is made after 15 min. Contour plots at various resolutions are shown in Fig. 3, and some summary statistics are provided in Table 3. There is quite a bit of variation between models for the precise solution at this point, as there is a lot of small scale structure forming, but our contour plots look broadly similar to those of Melvin et al. (2010). In particular, the density current has reached a very similar location, estimated as the maximum  $x$  coordinate over all cells where  $\Delta\theta < 0$ . We see that the dissipation of the minima of  $\Delta\theta$  is much weaker at the higher resolutions, and although there is still a substantial overshoot past the initial maxima of 0, this may be because the solution produces finer filaments of potential temperature at higher resolutions, which is more challenging for the advection scheme.

**Table 3** For the density current testcase: minimum, and maximum values of  $\Delta\theta = \theta - \theta_b$  measured at 15 min, together with the front location (estimated as the maximum  $x$  coordinate over all cells where  $\Delta\theta < 0$ )

$\Delta x$ (m)	$\Delta t$ (s)	$\Delta\theta_{max}$	$\Delta\theta_{min}$	Front location
800	4	1.3448	-11.6445	14,800
400	2	1.2293	-13.4614	15,000
200	1	0.9023	-16.3183	15,300
100	0.5	1.4226	-15.0101	15,250

The image has been cropped to focus on the right-propagating current

### 3.3 Flow over a mountain

This test problem simulating small amplitude lee waves generated by flow over a (small) mountain also has two versions, the nonhydrostatic flow regime version with velocity constrained to the  $x - z$  plane and consequently  $f = 0$ , and the hydrostatic flow regime version with 3D velocity and  $f = 10^{-4} \text{ s}^{-1}$ . Note that we are solving the nonhydrostatic equations in both versions, and the naming just describes the asymptotic flow regime and not the equations solved. The domain is given by  $L/2 \leq x \leq L/2$  and  $0 \leq z \leq H$  where  $L = 144,000 \text{ m}$  and  $H = 35,000 \text{ m}$  in the nonhydrostatic case and  $L = 240,000 \text{ m}$  and  $H = 50,000 \text{ m}$  in the hydrostatic case. In both cases there are periodic boundary conditions in the horizontal direction.

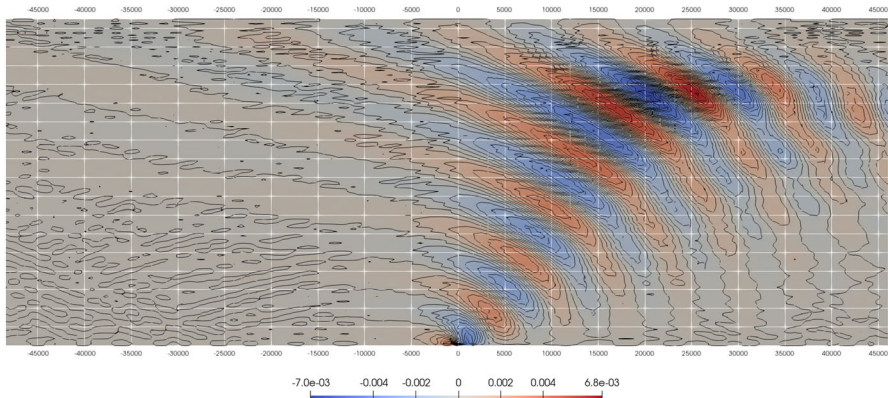
The mountain has a ‘‘Mount Agnesi’’ profile, with the bottom boundary moved to  $z_s(x) = \frac{a^2}{x^2+a^2}$ , giving a mountain of height  $1\text{m}$ . In our implementation, we use a simple terrain following mesh with the rectangular domain transformed according to  $(x, y, z) \mapsto (x, y, z + z_s(H - z)/H)$ . In the nonhydrostatic flow regime,  $a = 10,000 \text{ m}$ , and in the hydrostatic regime,  $a = 1000 \text{ m}$ .

In the nonhydrostatic flow regime, a stratified background flow is initialised according to the description of Sect. 3.1, with  $T_{surf} = 300 \text{ K}$ . In the hydrostatic flow regime, the stratification is isothermal, i.e. constant temperature at  $T = T_{surf} = 250 \text{ K}$ . This does still imply a varying potential temperature, with profile

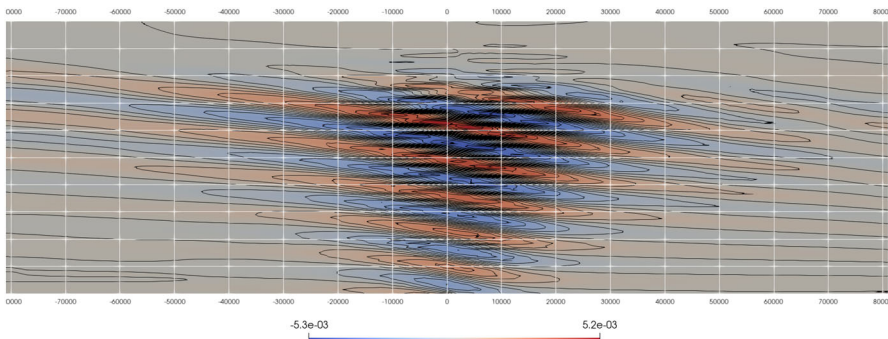
$$\theta = T_{surf} \exp\left(\frac{gz}{T_{surf}c_p}\right). \tag{20}$$

In both cases, the density is initialised by solving numerically for a hydrostatic profile, with boundary condition  $\Pi = 1$  at  $z = 0$ . This requires additional calculation because with the topography, the bottom boundary is not at  $z = 0$  everywhere. To address this, we calculate the boundary condition for  $\Pi$  at the top of the domain that produces the value  $\Pi = 1$  at  $z = 0$ , using a value on the bottom of the domain away from the mountain.

The nonhydrostatic test is initialised with a horizontal velocity  $u = 10 \text{ ms}^{-1}$  and the hydrostatic test is initialised with a horizontal velocity  $u = 20 \text{ ms}^{-1}$ . As for the gravity wave test, in the hydrostatic case, an additional forcing term is introduced to balance the Coriolis force, adding  $f \times (0, -20, 0)$  to the left hand side of Eq. (1). In both cases, the initial velocity is not compatible with the boundary condition at the mountain, so this causes a pressure wave propagating at ground level, radiating waves



**Fig. 4** Contours of vertical velocity for the nonhydrostatic mountain wave test at  $t = 9 \times 10^3$  s. Contour intervals are every  $5 \times 10^{-4} \text{ ms}^{-1}$ ; horizontal lines are plotted every 2000 m



**Fig. 5** Contours of vertical velocity for the hydrostatic mountain wave test at  $t = 1.5 \times 10^4$  s. Contour intervals are every  $5 \times 10^{-4} \text{ ms}^{-1}$ ; horizontal lines are plotted every 5000 m

that interfere with the stationary lee wave pattern that accumulates over time. This is often referred to as a test of robustness of the discretisation.

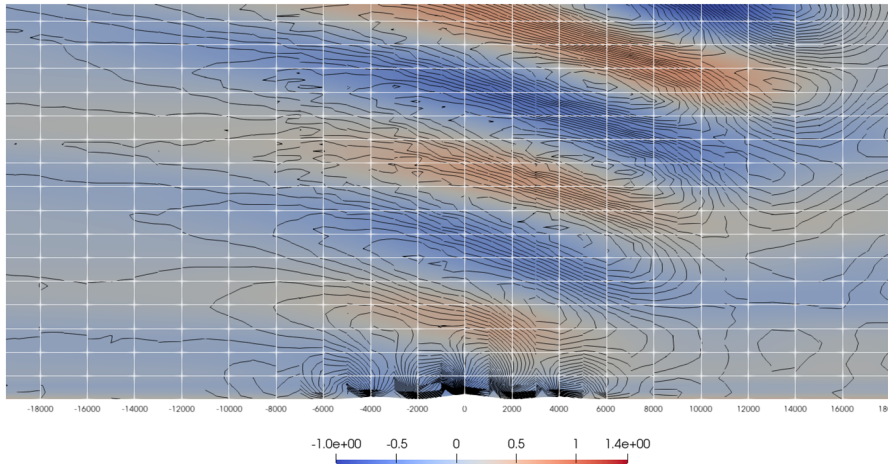
To prevent the lee waves (and the initial pressure waves caused by the sudden appearance of the mountain at time 0) reflecting off the top boundary, an absorbing term in the vertical velocity is added in the top layer, with profile

$$\mu(z) = \begin{cases} 0, & z < z_B, \\ \bar{\mu} \sin^2\left(\frac{\pi}{2}\left(\frac{z-z_B}{H-z_B}\right)\right), & z \geq z_B, \end{cases} \tag{21}$$

where  $\bar{\mu}$  is a constant and  $z_B$  is the height of the bottom of the absorbing layer. For the hydrostatic test,  $z_B = H - 2 \times 10^4 \text{ m} = 3 \times 10^4 \text{ m}$  and  $\bar{\mu}\Delta t = 0.3 \text{ s}$ . For the nonhydrostatic test,  $z_B = H - 10^4 \text{ m} = 2.5 \times 10^4 \text{ m}$  and  $\bar{\mu}\Delta t = 0.15 \text{ s}$ .

Contour plots of the vertical velocity are provided in Figs. 4 and 5, respectively. We observe good agreement with the solutions plotted in Melvin et al. (2010).





**Fig. 6** Contours of vertical velocity for the Schär mountain wave test at  $t = 1.8 \times 10^4$  s, with  $\Delta t = 8$  s. Contour intervals are every  $5 \times 10^{-2} \text{ ms}^{-1}$

### 3.4 Schär test

Schär et al. (2002) describe a more challenging mountain wave test with a mountain range orography that varies over multiple length scales, defined as

$$z_s(x) = h_m \exp\left(-\left(\frac{x}{a}\right)^2\right) \cos^2\left(\frac{\pi x}{\lambda}\right), \tag{22}$$

where  $h_m = 250$  m,  $\lambda = 4 \times 10^3$  m, and  $a = 5 \times 10^3$  m. The domain is given by  $-L/2 \leq x \leq L/2$  where  $L = 10^5$  m and  $0 \leq z \leq H = 3 \times 10^4$  m. The initial stratification is initialised in the same manner as for the nonhydrostatic mountain wave and we also add an absorbing term as described in Eq. 21, with  $z_B = H - 10^4 = 2 \times 10^4$  m and  $\bar{\mu} \Delta t = 1.2$  s. The velocity is initially horizontal with  $u = 10 \text{ ms}^{-1}$ . Following Melvin et al. (2010), we ran the test with two values of  $\Delta t$ , 8 s and 40 s. As might be expected from a solution that has almost reached a steady state for the lee wave pattern and the fact that we are not using a splitting method, the numerical solutions obtained were indistinguishable. The larger timestep requires more linear solver iterations, as displayed in Table 2. This is offset by the larger timestep, so the time to solution is similar (shorter for the larger timestep, in fact). This reflects the fact that our columnwise preconditioner produces mesh independent convergence rates when applied to the linearisation about the state of rest, but has a dependency on Courant number in general. The solutions fit within the range of results obtained by others e.g. Straka et al. (1993); Giraldo and Restelli (2008); Bendall et al. (2020), given the turbulent nature of the solution (Fig. 6).

## 4 Summary and outlook

In this paper we presented a compatible finite element discretisation for the compressible Euler equations, and demonstrated numerical robustness using a standard suite of vertical slice tests for numerical weather prediction. In all cases the results are very similar to published results. In future work we will demonstrate the discretisation in the fully three dimensional setting on the sphere.

One novel feature of our approach is a monolithic approach to solving a fully implicit system. This approach shows promise for producing robust timestepping approaches. In experiments with the shallow water equations we have used the additive Schwarz approach as a smoother for a multigrid scheme, which has led to faster convergence of the iterative solver. We were unable to do that in this work because Firedrake does not currently support mesh hierarchies on periodic meshes which are required for this suite of tests. We will rectify this in further work.

One reason for our interest in fully implicit methods is that these are required for the time parallel algorithms that we are currently developing. It is also interesting to consider Rosenbrock methods that only require the solution of linear systems (as opposed to the nonlinear systems coming from the implicit midpoint rule) linearised about the state at the start of the timestep. In particular, we plan to experiment with a Rosenbrock version of the TR-BDF2 timestepping scheme, previously applied to numerical weather prediction in Tumolo and Bonaventura (2015).

**Funding** The authors would like to acknowledge funding from UKRI and The Met Office (SPF ExCALIBUR EX20-8), the EPSRC PRISM platform grants EP/L000407/1 and EP/R029423/1, EPSRC grant EP/R029628/1 and NERC grants NE/K006789/1, NE/I02013X/1, NE/R008795/1 and NE/M013634/1.

## Declarations

**Conflict of interest** The authors have no relevant financial or non-financial interests to disclose.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

- Adler, J.H., Benson, T.R., Cyr, E.C., Farrell, P.E., MacLachlan, S.P., Tuminaro, R.S.: Monolithic multigrid methods for magnetohydrodynamics. *SIAM J. Sci. Comput.* **43**, S70–S91 (2021)
- Alnæs, M.S., Logg, A., Ølgaard, K.B., Rognes, M.E., Wells, G.N.: Unified form language: a domain-specific language for weak formulations of partial differential equations. *ACM Trans. Math. Softw. (TOMS)* **40**, 1–37 (2014)
- Arakawa, A., Lamb, V.R.: A potential enstrophy and energy conserving scheme for the shallow water equations. *Mon. Weather Rev.* **109**, 18–36 (1981)

- Balay, S., Abhyankar, S., Adams, M.F., Brown, J., Brune, P., Buschelman, K., Dalcin, L., Dener, A., Eijkhout, V., Gropp, W., et al.: PETSc Users Manual (Rev. 3.13). Technical Report. Argonne National Lab.(ANL), Argonne, IL (United States) (2020)
- Bauer, W., Cotter, C.J.: Energy-entropy conserving compatible finite element schemes for the rotating shallow water equations with slip boundary conditions. *J. Comput. Phys.* **373**, 171–187 (2018)
- Bendall, T.M., Gibson, T.H., Shipton, J., Cotter, C.J., Shipway, B.: A compatible finite-element discretisation for the moist compressible Euler equations. *Q. J. R. Meteorol. Soc.* **146**, 3187–3205 (2020)
- Bercea, G.T., McRae, A.T., Ham, D.A., Mitchell, L., Rathgeber, F., Nardi, L., Luporini, F., Kelly, P.H.: A structure-exploiting numbering algorithm for finite elements on extruded meshes, and its performance evaluation in Firedrake. *Geosci. Model Dev.* **9**, 3803–3815 (2016)
- Betteridge, J.D., Cotter, C.J., Gibson, T.H., Griffith, M.J., Melvin, T., Müller, E.H.: Hybridised multigrid preconditioners for a compatible finite element dynamical core. *arXiv preprint arXiv:2210.11797* (2022)
- Burman, E.: A unified analysis for conforming and nonconforming stabilized finite element methods using interior penalty. *SIAM J. Numer. Anal.* **43**, 2012–2033 (2005)
- Burman, E., Ern, A.: Continuous interior penalty hp-finite element methods for advection and advection-diffusion equations. *Math. Comput.* **76**, 1119–1140 (2007)
- Cockburn, B., Kanschä, G., Schötzau, D.: A note on discontinuous Galerkin divergence-free solutions of the Navier–Stokes equations. *J. Sci. Comput.* **31**, 61–73 (2007)
- Cotter, C.J., Shipton, J.: Mixed finite elements for numerical weather prediction. *J. Comput. Phys.* **231**, 7076–7091 (2012)
- Cotter, C.J., Thuburn, J.: A finite element exterior calculus framework for the rotating shallow-water equations. *J. Comput. Phys.* **257**, 1506–1526 (2014)
- Danilov, S.: On utility of triangular c-grid type discretization for numerical modeling of large-scale ocean flows. *Ocean Dyn.* **60**, 1361–1369 (2010)
- Dubos, T., Dubey, S., Tort, M., Mittal, R., Meurdesoif, Y., Hourdin, F.: Dynamico-1.0, an icosahedral hydrostatic dynamical core designed for consistency and versatility. *Geosci. Model Dev.* **8**, 3131–3150 (2015)
- Eldred, C., Dubos, T., Kritsikis, E.: A quasi-Hamiltonian discretization of the thermal shallow water equations. *J. Comput. Phys.* **379**, 1–31 (2019)
- Farrell, P.E., Mitchell, L., Wechsung, F.: An augmented lagrangian preconditioner for the 3d stationary incompressible Navier–Stokes equations at high reynolds number. *SIAM J. Sci. Comput.* **41**, A3073–A3096 (2019)
- Gassmann, A.: A global hexagonal c-grid non-hydrostatic dynamical core (icon-iap) designed for energetic consistency. *Q. J. R. Meteorol. Soc.* **139**, 152–175 (2013)
- Gawlik, E.S., Gay-Balmaz, F.: A conservative finite element method for the incompressible Euler equations with variable density. *J. Comput. Phys.* **412**, 109439 (2020)
- Gawlik, E.S., Gay-Balmaz, F.: A finite element method for mhd that preserves energy, cross-helicity, magnetic helicity, incompressibility, and  $\text{div } \mathbf{b} = 0$ . *J. Comput. Phys.* **450**, 110847 (2022)
- Giraldo, F.X., Restelli, M.: A study of spectral element and discontinuous Galerkin methods for the Navier–Stokes equations in nonhydrostatic mesoscale atmospheric modeling: Equation sets and test cases. *J. Comput. Phys.* **227**, 3849–3877 (2008)
- Guerra, J.E., Ullrich, P.A.: A high-order staggered finite-element vertical discretization for non-hydrostatic atmospheric models. *Geosci. Model Dev.* **9**, 2007–2029 (2016)
- Laakmann, F., Farrell, P.E., Mitchell, L.: An augmented Lagrangian preconditioner for the magnetohydrodynamics equations at high Reynolds and coupling numbers. *SIAM J. Sci. Comput.* **44**, B1018–B1044 (2022)
- Lee, D., Palha, A.: A mixed mimetic spectral element model of the rotating shallow water equations on the cubed sphere. *J. Comput. Phys.* **375**, 240–262 (2018)
- Lee, D., Palha, A.: A mixed mimetic spectral element model of the 3d compressible Euler equations on the cubed sphere. *J. Comput. Phys.* **401**, 108993 (2020)
- McRae, A.T., Bercea, G.T., Mitchell, L., Ham, D.A., Cotter, C.J.: Automated generation and symbolic manipulation of tensor product finite elements. *SIAM J. Sci. Comput.* **38**, S25–S47 (2016)
- McRae, A.T., Cotter, C.J.: Energy- and entropy-conserving schemes for the shallow-water equations, based on mimetic finite elements. *Q. J. R. Meteorol. Soc.* **140**, 2223–2234 (2014)

- Melvin, T., Benacchio, T., Shipway, B., Wood, N., Thuburn, J., Cotter, C.: A mixed finite-element, finite-volume, semi-implicit discretization for atmospheric dynamics: cartesian geometry. *Q. J. R. Meteorol. Soc.* **145**, 2835–2853 (2019)
- Melvin, T., Dubal, M., Wood, N., Staniforth, A., Zerroukat, M.: An inherently mass-conserving iterative semi-implicit semi-Lagrangian discretization of the non-hydrostatic vertical-slice equations. *Q. J. R. Meteorol. Soc.: J. Atmos. Sci. Appl. Meteorol. Phys. Oceanogr.* **136**, 799–814 (2010)
- Natale, A., Cotter, C.J.: Scale-selective dissipation in energy-conserving finite-element schemes for two-dimensional turbulence. *Q. J. R. Meteorol. Soc.* **143**, 1734–1745 (2017)
- Natale, A., Cotter, C.J.: A variational finite-element discretization approach for perfect incompressible fluids. *IMA J. Numer. Anal.* **38**, 1388–1419 (2018)
- Natale, A., Shipton, J., Cotter, C.J.: Compatible finite element spaces for geophysical fluid dynamics. *Dyn. Stat. Clim. Syst.* **1**, 1 (2016)
- Rathgeber, F., Ham, D.A., Mitchell, L., Lange, M., Luporini, F., McRae, A.T., Bercea, G.T., Markall, G.R., Kelly, P.H.: Firedrake: automating the finite element method by composing abstractions. *ACM Trans. Math. Softw. (TOMS)* **43**, 1–27 (2016)
- Rognes, M.E., Kirby, R.C., Logg, A.: Efficient assembly of  $H(\text{div})$  and  $H(\text{curl})$  conforming finite elements. *SIAM J. Sci. Comput.* **31**, 4130–4151 (2010)
- Schär, C., Leuenberger, D., Fuhrer, O., Lüthi, D., Girard, C.: A new terrain-following vertical coordinate formulation for atmospheric prediction models. *Mon. Weather Rev.* **130**, 2459–2480 (2002)
- Sergeev, D.E., Mayne, N.J., Bendall, T., Boutle, I.A., Brown, A., Kavcic, I., Kent, J., Kohary, K., Manners, J., Melvin, T., et al.: Simulations of idealised 3d atmospheric flows on terrestrial planets using LFRIC-atmosphere. *arXiv preprint arXiv:2306.03614* (2023)
- Shipton, J., Gibson, T.H., Cotter, C.J.: Higher-order compatible finite element schemes for the nonlinear rotating shallow water equations on the sphere. *J. Comput. Phys.* **375**, 1121–1137 (2018)
- Skamarock, W.C., Klemp, J.B.: Efficiency and accuracy of the Klemp–Wilhelmson time-splitting technique. *Mon. Weather Rev.* **122**, 2623–2630 (1994)
- Staniforth, A., Thuburn, J.: Horizontal grids for global weather and climate prediction models: a review. *Q. J. R. Meteorol. Soc.* **138**, 1–26 (2012)
- Straka, J.M., Wilhelmson, R.B., Wicker, L.J., Anderson, J.R., Droegemeier, K.K.: Numerical solutions of a non-linear density current: a benchmark solution and comparisons. *Int. J. Numer. Meth. Fluids* **17**, 1–22 (1993)
- Taylor, M.A., Guba, O., Steyer, A., Ullrich, P.A., Hall, D.M., Eldred, C.: An energy consistent discretization of the nonhydrostatic equations in primitive variables. *J. Adv. Model. Earth Syst.* **12**, e2019MS001783 (2020)
- Thuburn, J., Cotter, C., Dubos, T.: A mimetic, semi-implicit, forward-in-time, finite volume shallow water model: comparison of hexagonal-icosahedral and cubed-sphere grids. *Geosci. Model Dev.* **7**, 909–929 (2014)
- Thuburn, J., Cotter, C.J.: A framework for mimetic discretization of the rotating shallow-water equations on arbitrary polygonal grids. *SIAM J. Sci. Comput.* **34**, B203–B225 (2012)
- Tumolo, G., Bonaventura, L.: A semi-implicit, semi-Lagrangian discontinuous Galerkin framework for adaptive numerical weather prediction. *Q. J. R. Meteorol. Soc.* **141**, 2582–2601 (2015)
- Wimmer, G.A., Cotter, C.J., Bauer, W.: Energy conserving upwinded compatible finite element schemes for the rotating shallow water equations. *J. Comput. Phys.* **401**, 109016 (2020)
- Wimmer, G.A., Cotter, C.J., Bauer, W.: Energy conserving SUPG methods for compatible finite element schemes in numerical weather prediction. *SMAI J. Comput. Math.* **7**, 267–300 (2021)
- Wood, N., Staniforth, A., White, A., Allen, T., Diamantakis, M., Gross, M., Melvin, T., Smith, C., Vosper, S., Zerroukat, M., et al.: An inherently mass-conserving semi-implicit semi-Lagrangian discretization of the deep-atmosphere global non-hydrostatic equations. *Q. J. R. Meteorol. Soc.* **140**, 1505–1520 (2014)
- Yamazaki, H., Shipton, J., Cullen, M.J., Mitchell, L., Cotter, C.J.: Vertical slice modelling of nonlinear Eady waves using a compatible finite element method. *J. Comput. Phys.* **343**, 130–149 (2017)