Adaptively implicit MPDATA advection for arbitrary Courant numbers and Meshes

Article

Published Version

Weller, H. ORCID: https://orcid.org/0000-0003-4553-7082, Woodfield, J., Kühnlein, C. and Smolarkiewicz, P. K. (2023) Adaptively implicit MPDATA advection for arbitrary Courant numbers and Meshes. Quarterly Journal of the Royal Meteorological Society, 149 (751). pp. 369-388. ISSN 1477-870X doi: 10.1002/qj.4411 Available at https://centaur.reading.ac.uk/108775/

To link to this article DOI: http://dx.doi.org/10.1002/qj.4411

Publisher: Royal Meteorological Society

Received: 27 July 2022 | Revised: 17 October 2022 | Accepted: 5 November 2022 | Published on: 19 December 2022

DOI: 10.1002/qj.4411

Quarterly Journal of the Royal Meteorological Society RMetS

RESEARCH ARTICLE

Adaptively implicit MPDATA advection for arbitrary Courant numbers and meshes

Hilary Weller[1] | **James Woodfield**[2] | **Christian Kühnlein**[3] | **Piotr K. Smolarkiewicz**[4]

[1]Meteorology, University of Reading, Reading, UK

[2]Mathematics, University of Reading, Reading, UK

[3]European Centre for Medium-Range Weather Forecasts, Bonn, Germany

[4]National Center for Atmospheric Research, Boulder, CO, USA

Correspondence
H. Weller, Meteorology, University of Reading, Reading, RG6 6BB, UK.
Email: h.weller@reading.ac.uk

Abstract

Advection schemes with time step restrictions are widely used in weather and climate models. This can lead to instability in the presence of high flow speeds (relative to mesh spacing) such as occurs in convective updraughts, regions of mesh convergence, or where the winds are unusually high. An adaptively implicit advection scheme is proposed that treats advection implicitly only where the Courant number is high. Flux correction to ensure monotonicity is adapted to work with implicit time stepping. A version of the multidimensional positive-definite advection transport algorithm MPDATA is derived with an anti-diffusive flux compensating truncation errors of off-centred implicit time stepping. The anti-diffusive flux is gradually reduced as Courant numbers increase above 2 in order to maintain stability at the expense of second-order accuracy at high Courant numbers. Results of two-dimensional advection by deformational flow are presented on various meshes of the sphere. Stability and second-order accuracy are maintained when the Courant number is over 100 in a small region, when strong wind crosses the poles of a rotated latitude–longitude mesh. Good solutions are also obtained on a skipped latitude–longitude mesh, a cubed sphere, and hexagonal meshes. Accuracy reverts to first order when Courant numbers are large over a large fraction of the domain.

KEYWORDS

advection, Courant number, implicit, monotonicity, stable, transport

1 | INTRODUCTION

1.1 | Motivation

Time step restrictions based on advection have always posed a problem for models of the atmosphere. The Courant–Friedrichs–Lewy condition states that explicit Eulerian advection schemes will have time step restrictions based on the size of the spatial discretisation increments and the flow speed. Typically, this means that explicit schemes cannot run with an effective Courant number >1. This is related to saying that an advected quantity cannot be moved by more than one mesh cell (or grid box or element) in one time step.

Courant–Friedrichs–Lewy constraints can be cruel where spatial resolution is higher than it needs to be for accuracy, such as near the poles of a latitude–longitude mesh. An early workaround was to use polar filtering; that is, artificially removing oscillations near the poles where mesh lines converge (Cullen and Davies, 1991). However, polar filtering led to parallel scaling bottlenecks and mistrust of solutions near the poles in models of the atmosphere. The UK Met Office gained accuracy and efficiency by replacing their model employing polar filtering with a semi-implicit, semi-Lagrangian (SISL) model to avoid time step restrictions on a latitude–longitude mesh (Davies *et al.*, 2005). SISL eases time step constraints by treating acoustic and gravity waves implicitly, and the advection is solved with the semi-Lagrangian method, which is stable, accurate for smooth flows, and efficient with long time steps but is not conservative. The lack of conservation is regarded as inadequate for climate modelling and is associated with spurious features such as eternal fountains, which involve a positive feedback loop that creates moisture in convectively unstable columns (Zerroukat and Allen, 2020).

The requirement to run stably with large Courant numbers is less severe now that models of the global atmosphere have largely moved away from latitude–longitude meshes in favour of quasi-uniform meshes, such as the cubed sphere and icosahedral meshes (Ullrich *et al.*, 2017). However, the problem remains severe in the vertical direction, where mesh spacing can be fine and large vertical velocities can occur when atmospheric convection is resolved. Solutions in the vertical direction can exploit the fact that, even on a horizontally unstructured mesh, grids in the vertical are structured and one-dimensional, meaning that conservative (semi-)Lagrangian methods can be made to work efficiently. An example is the use of Lagrangian floating levels (Lin, 2004) that are conservatively mapped back to the fixed Eulerian mesh. The Lagrangian floating levels technique is equivalent to a conservative semi-Lagrangian (Harris *et al.*, 2011) and can be made efficient for large Courant numbers in one dimension and consequently can work on tensor-product (logically rectangular) meshes (e.g., Leonard *et al.*, 1996).

Returning to the problem of large Courant numbers in the horizontal, some conservative, flux-form semi-Lagrangian schemes have been developed for arbitrary meshes (e.g., Miura, 2007) but only work efficiently for large Courant numbers on tensor-product grids. The conservative semi-Lagrangian scheme is related to the arbitrary Lagrangian–Eulerian method (e.g., Hirt *et al.*, 1997), which solves equations in a Lagrangian frame and then remaps the solution back to the original mesh. However, arbitrary Lagrangian–Eulerian methods suffer from time step restrictions associated with avoiding mesh tangling.

Local time stepping (LTS) involves taking multiple time steps in some regions of the domain to avoid large Courant numbers. Thus, efficiency can be improved, since some regions can use large time steps. This can be useful in variable resolution models (e.g., Dumbser *et al.*, 2007). The variability of Courant numbers is particularly large when small elements are used to improve accuracy in regions where flow rates are high; for example, representing narrow passages in coastal ocean models (e.g., Dawson *et al.*, 2013). LTS has been used for atmospheric modelling (Baldauf, 2019), leading to some improved efficiency. However, owing to spatially and temporally variable time step (hence computational effort), LTS is associated with challenging load balancing issues, and efficiency gains can be modest if only a few large cells use the large time steps.

Returning to the difficulties in allowing long time steps while maintaining exact local conservation on arbitrary meshes, it is worth considering implicit time stepping for advection. The aim of this paper is to present an advection scheme with the following properties:

1. Stable for large Courant numbers.
2. Applicable on arbitrary meshes.
3. Locally conservative to machine precision.
4. Similar cost and accuracy to explicit schemes when the Courant number is <1.
5. At least first-order accurate where the Courant number is large.
6. Options to be monotonic, bounded, or sign preserving.
7. Multi-tracer efficient.
8. Good parallel scaling.

We define a bounded advection scheme as one that does not produce solution values outside specified bounds; for example, $[0, 1]$. The multidimensional positive definite advection transport algorithm (MPDATA) is sign preserving (positive definite), which means solutions that start in $[0, \infty)$ stay in $[0, \infty)$. A monotonic advection scheme does not generate new spurious extrema or amplify existing extrema. This is desirable, as it implies boundedness and also guarantees stability. The question of how to define monotonicity of an implicit scheme is addressed in Appendix A.

We propose the use herein of adaptively implicit time stepping combined with MPDATA to create a scheme with options to be either monotonic, bounded, sign preserving, or just stable for all Courant numbers. The aim is not to produce a scheme that is accurate for large Courant numbers in a situation where, for example, resolution of extreme weather is crucial. The aim is to produce a scheme that is robust when locally high Courant numbers threaten stability and when high resolution in time is not needed for the features producing locally high Courant numbers. The methods for combining implicit and explicit time stepping and the method for achieving monotonicity should translate to other explicit advection schemes. They are also suitable for horizontal and vertical discretisation.

1.2 | Background on MPDATA

The MPDATA scheme for the solution of the advection equation was introduced by Smolarkiewicz (1983; 1984). The basic principle of MPDATA is as follows: apply the forward-in-time, first-order upwind scheme based on the physical flow velocity, followed by a corrective upwind step with a pseudo-velocity that is designed to compensate the spatial and temporal truncation errors of the preceding step to at least second order. Owing to the repeated application of the upwind scheme, MPDATA provides strict sign preservation of the transported field and a small phase error. The standard MPDATA is an explicit flux-form Eulerian scheme and stable for an advective Courant number <1. Various MPDATA extensions, including some for entire flow solvers, have been developed over the years. Smolarkiewicz and Margolin (1998) review the schemes with a focus on structured grids.

MPDATA options of particular interest to this paper are the infinite-gauge variant, a linear scheme that is not sign preserving (Smolarkiewicz and Clark, 1986; Smolarkiewicz and Margolin, 1998), and the extension to enforce solution monotonicity (Smolarkiewicz and Grabowski, 1990) by means of flux-corrected transport (FCT; Zalesak, 1979). Smolarkiewicz and Szmelter (2005) extended MPDATA to fully unstructured meshes while retaining second-order accuracy and the other favourable properties of the scheme. Kühnlein and Smolarkiewicz (2017) formulate the MPDATA pseudo-velocity based solely on face-normal fluxes, which facilitated integration of compressible partial differential equations on arbitrary unstructured meshes. Whereas the standard MPDATA scheme is fully multidimensional, Kühnlein *et al.* (2019) use a horizontal–vertical second-order accurate Strang-split integration based on MPDATA that permits larger time steps and also enables more targeted schemes in the different coordinate directions of the global atmospheric model.

1.3 | Background on implicit advection schemes

Implicit time stepping is ubiquitous in atmospheric modelling for solving the terms of the equations of motion responsible for fast waves, such as gravity and acoustic waves. However, implicit time stepping has rarely been used for advection in atmosphere and ocean modelling. Implicit time stepping for advection in the mathematics and engineering literature will be discussed first, and then we will return to its uses to date in atmosphere and ocean modelling.

Implicit time stepping for advection has a severe order barrier; no implicit method exists with order greater than one that is monotonic for all time steps (Gottlieb *et al.*, 2001). Higher order implicit multistage (Runge–Kutta) and multistep schemes exist that are unconditionally linearly stable, but if we additionally require monotonicity (no new spurious extrema generated) then higher order implicit methods have time step restrictions, characterised by the radius of monotonicity. The nonlinearity required for the spatial discretisation to remain monotone becomes even more challenging with implicit time stepping because the required nonlinearity is approximated by linear solution strategies. This was explored by Yee and co-workers (Yee *et al.*, 1985; Yee, 1987), where a high-order spatial discretisation was combined with the backward Euler scheme, but monotonicity was only achieved at the price of mass conservation. May and Berger (2017) used FCT (Zalesak, 1979) to improve temporal accuracy in implicitly solved small cells without generating new extrema.

In atmospheric modelling, implicit time stepping for advection has been used for vertical advection (Baldauf *et al.*, 2011) and to treat small, cut-cells stably at modest time steps (e.g., Jebens *et al.*, 2011). Wicker and Skamarock (2020) and Li and Zhang (2022) use adaptively implicit vertical transport to treat isolated strong updraughts stably, avoiding the order barrier by limiting order of accuracy to first (upwind) wherever implicit time stepping is used. The lack of accuracy was not considered problematic because of the sparsity of the use of implicit advection, although Li and Zhang (2022) describe the implicit advection as being more diffusive than explicit advection. Chen *et al.* (2017) compared implicit advection with dimensionally split, flux-form semi-Lagrangian advection and found that the dimensionally split scheme was more accurate and more efficient than implicit advection for all Courant numbers. However, this was not a like-for-like comparison; the dimensionally split scheme was a higher order accurate scheme, limited to tensor product meshes, and suffered from mesh imprinting errors on distorted meshes.

1.4 | Outline

The description of the adaptively implicit MPDATA in Section 2 starts with an alternative formulation of the standard explicit MPDATA on an unstructured, centroidal mesh, without coordinate transforms. This description is then extended to the implicit case including a description of how the explicit and implicit schemes are blended to ensure efficiency and stable solutions, how the infinite-gauge variant is used with the implicit scheme, and how FCT can be used with an implicit scheme. The description is general for one, two, and three dimensions, but in this paper, for simplicity of the presentation, we do not include the terms for divergent velocity fields. The numerical results in Section 3 start in one dimension, comparing an explicit scheme on a uniform grid with an adaptively implicit scheme on a non-uniform grid. The remainder of the results are of deformational flow on the surface of the sphere, demonstrating convergence and monotonicity for a wide range of Courant numbers. The method described is tested for either horizontal or vertical discretisation in this paper, but it should also be applicable in three dimensions. The adaptively implicit time stepping with first-order spatial discretisation is proved to be bounded in Appendix A, and a one-dimensional version of the adaptively implicit scheme is proved to be stable for all Courant numbers in Appendix B.

2 | AN ADAPTIVELY IMPLICIT MPDATA

The version of explicit MPDATA and the adaptively implicit MPDATA defined here are implemented using the OpenFOAM library (https://openfoam.org/) using standard OpenFOAM operators and linear equation solvers. The code is available as part of the AtmosFOAM repository (https://github.com/AtmosFOAM/) compiled with OpenFOAM7.

2.1 | Explicit MPDATA on an arbitrary mesh

The description of the explicit scheme is consistent with basic MPDATA principles (e.g., Smolarkiewicz and Szmelter, 2005) but introduces a novel derivation based on a flux-form semi-Lagrangian method and assuming an arbitrary, centroidal mesh in Cartesian coordinates rather than using coordinate transforms. We will describe MPDATA for solving the linear advection equation for advected quantity ψ with velocity field $\mathbf{u}$:

$$\frac{\partial \psi}{\partial t} + \nabla \cdot (\mathbf{u}\psi) = 0. \tag{1}$$

This is solved using Gauss's divergence theorem on an arbitrary mesh to go from time t^n to t^{n+1} a time step Δt apart:

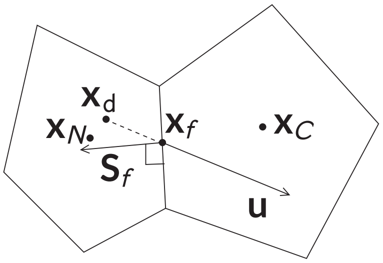


FIGURE 1 Two cells in an arbitrary mesh. $\mathbf{x}_C$ and $\mathbf{x}_N$ are the cell centres (centroids) of cell C and its neighbour N over face f, and $\mathbf{x}_f$ is the face centre. $\mathbf{S}_f$ is the face area vector, normal to face f with magnitude equal to the face area. $\mathbf{u}$ is the velocity, and $\mathbf{x}_\mathrm{d}$ is the departure point for face f at $t^{n+1/2}$ (i.e., the centre of the volume swept through the face between t^n and t^{n+1})

$$\psi_\mathrm{c}^{n+1} = \psi_\mathrm{c}^{n} - \frac{\Delta t}{\mathcal{V}_\mathrm{c}} \sum_{f \in C} \psi_f^{n+1/2} U_f, \tag{2}$$

where ψ_c is the cell mean value of ψ in cell C, $\mathcal{V}_\mathrm{c}$ is the volume of cell C, $f \in C$ are the faces of cell C, $\psi_f^{n+1/2}$ is the value of ψ at face f at $t^{n+1/2}$, $\mathbf{u}_f$ is the velocity at face f, and $\mathbf{S}_f$ is the face area vector – the outward-pointing vector normal to face f with magnitude equal to the area of face f (Figure 1). $U_f = \mathbf{u}_f \cdot \mathbf{S}_f$ is the volume flux over face f. In this derivation of MPDATA, ψ_f is evaluated at the departure point of the face centre at t^n. The departure point $\mathbf{x}_\mathrm{d}$ is the centre of the volume that is swept through the face between t^n and t^{n+1}. This is approximated by the point a distance $\mathbf{u}\Delta t/2$ upstream of the face centre:

$$\mathbf{x}_\mathrm{d} = \mathbf{x}_f - \frac{\Delta t}{2}\mathbf{u}_f + O(\Delta t)^2, \tag{3}$$

where $\mathbf{x}_f$ is the face centre (see Figure 1). The velocity, $\mathbf{u}_f$, is evaluated at $t^{n+1/2}$ at the face centre. In this paper, we consider passive advection of ψ with a prescribed non-divergent wind field. In a dynamical model, $\mathbf{u}_f^{n+1/2}$ would be evaluated from velocities at known positions and times and non-divergent winds would be treated as in Smolarkiewicz and Margolin (1998), Smolarkiewicz (2006), Kühnlein *et al.* (2012), and Kühnlein and Smolarkiewicz (2017).

The dependent variable ψ is evaluated at the departure point, $\mathbf{x}_\mathrm{d}$, using the upwind cell centre value of ψ, the gradient of ψ at the face centre, and the velocity divergence at the face centre (assumed zero):

$$\psi_f^{n+1/2} = \psi_\mathrm{d}^n = \psi_\mathrm{up}^n + (\mathbf{x}_\mathrm{d} - \mathbf{x}_\mathrm{up}) \cdot \nabla \psi^n - \cancelto{0}{\frac{\Delta t}{2}\psi_\mathrm{up}^n \nabla \cdot \mathbf{u}} + O(\Delta s^2, \Delta t^2), \tag{4}$$

where ψ_{up} is the value of ψ in the cell upwind of face f, Δs is the cell centre to cell centre distance, $\mathbf{x}_{\text{up}}$ is the location of the upwind cell centre, and $\mathbf{x}_{\text{down}}$ is the location of the downwind cell centre (which will be used later). Equations (3) and (4) are substituted in to Equation (2) to give a scheme that is second-order accurate in space and time but not sign preserving, equivalent to a Lax–Wendroff scheme:

$$\begin{aligned}\psi_{\text{c}}^{n+1} = \psi_{\text{c}}^{n} &- \underbrace{\frac{\Delta t}{\mathcal{V}_{\text{c}}}\sum_{f\in C}\psi_{\text{up}}^{n}U_f}_{\text{explicit upwind}} - \underbrace{\frac{\Delta t}{\mathcal{V}_{\text{c}}}\sum_{f\in C}(\mathbf{x}_f-\mathbf{x}_{\text{up}})\cdot\nabla\psi^{n}U_f}_{\text{spatial correction}}\\ &+ \underbrace{\frac{\Delta t^2}{2\mathcal{V}_{\text{c}}}\sum_{f\in C}\mathbf{u}_f\cdot(\nabla\psi^{n})_f U_f}_{\text{temporal correction}} \qquad (5)\\ &+ \Delta t\; O(\Delta s^2, \Delta t^2).\end{aligned}$$

The explicit upwind scheme is monotone. In order for the spatial and temporal correction terms to be sign preserving they are written as explicit upwind advection using an anti-diffusive flux $V_f = \mathbf{v}_f\cdot\mathbf{S}_f$. The anti-diffusive flux is divergent, so new extrema are not prevented. The use of the anti-diffusive flux transforms the scheme from Lax–Wendroff to MPDATA and is written in two stages:

$$\text{explicit upwind step: } \psi_{\text{c}}^{1} = \psi_{\text{c}}^{n} - \frac{\Delta t}{\mathcal{V}_{\text{c}}}\sum_{f\in C}\psi_{\text{up}}^{n}U_f, \qquad (6)$$

$$\text{explicit correction: } \psi_{\text{c}}^{n+1} = \psi_{\text{c}}^{1} - \frac{\Delta t}{\mathcal{V}_{\text{c}}}\sum_{f\in C}\psi_{\text{vup}}^{1}V_f, \qquad (7)$$

where

$$\begin{aligned}V_f &= \mathbf{v}_f\cdot\mathbf{S}_f\\ &= \frac{U_f}{\psi}\left[(\mathbf{x}_f-\mathbf{x}_{\text{up}})\cdot\nabla\psi - \frac{\Delta t}{2}\mathbf{u}_f\cdot(\nabla\psi)_f\right] \qquad (8)\end{aligned}$$

and where ψ_{vup} is ψ in the upwind cell where the upwind direction is defined by the sign of V_f. As ψ is always positive, there is no ambiguity in the sign of V_f. The anti-diffusive flux V_f can be calculated iteratively, using first ψ^1 and then subsequent iterations use the most up-to-date version of ψ^{n+1} and V_f. All simulations in this paper use one iteration per time step. Equation (8) is a continuous version of the expression for the anti-diffusive velocity in Smolarkiewicz and Szmelter (2005, eq. 13). The discretisations of gradients and divergences described here are similar to those of Smolarkiewicz and Szmelter (2005).

The first term of Equation (8) is discretised by assuming that $\mathbf{x}_f - \mathbf{x}_{\text{up}} = \frac{1}{2}(\mathbf{x}_{\text{down}} - \mathbf{x}_{\text{up}})$ and preventing division by zero:

$$\frac{(\mathbf{x}_f-\mathbf{x}_{\text{up}})\cdot\nabla\psi}{\psi} \approx \frac{\psi_{\text{down}}-\psi_{\text{up}}}{\psi_{\text{down}}+\psi_{\text{up}}+\varepsilon}, \qquad (9)$$

which is second-order accurate only on non-skew, uniform grids. The results presented in Section 3 use $\varepsilon = 10^{-16}$. The second term of Equation (8) is discretised using a second-order, least-squares approximation for $\nabla\psi$ in cells:

$$\nabla\psi = \sum_{f\in C}\mathbf{g}_f(\psi_N - \psi_{\text{c}}), \qquad (10)$$

where ψ_N is ψ in the neighbour of cell C across face f and where $\mathbf{g}_f$ is a vector calculated for each face of cell C based entirely on the local mesh geometry:

$$\mathbf{g}_f = (1-w_f)\frac{|\mathbf{S}_f|}{|\mathbf{x}_N-\mathbf{x}_C|^2}D_C^{-1}(\mathbf{x}_N-\mathbf{x}_C), \qquad (11)$$

where

$$w_f = \frac{|\mathbf{S}_f\cdot(\mathbf{x}_N-\mathbf{x}_f)|}{|\mathbf{S}_f\cdot(\mathbf{x}_N-\mathbf{x}_f)| + |\mathbf{S}_f\cdot(\mathbf{x}_f-\mathbf{x}_C)|}, \quad \text{(interpolation weights)}$$

and

$$D_C = \sum_{f\in C}(1-w_f)\frac{|\mathbf{S}_f|}{|\mathbf{x}_N-\mathbf{x}_C|^2}(\mathbf{x}_N-\mathbf{x}_C)(\mathbf{x}_N-\mathbf{x}_C)^{\text{T}}.$$

This is the least-squares gradient implemented in the OpenFOAM library. Cell centre gradients are then linearly interpolated onto face centres, denoted $(\nabla\psi)_f$. The component in the $\mathbf{x}_N - \mathbf{x}_C$ direction is corrected using the compact gradient:

$$(\nabla\psi)_f\cdot(\mathbf{x}_N-\mathbf{x}_C) = \psi_N - \psi_{\text{c}}. \qquad (12)$$

2.1.1 | Sign preservation

Given the definition of the Courant number, c, on an arbitrary mesh,

$$c = \frac{1}{2}\frac{\Delta t}{\mathcal{V}_{\text{c}}}\sum_f |U_f|, \qquad (13)$$

Smolarkiewicz and Szmelter (2005) showed that explicit upwind is monotonic for non-divergent velocity fields when $c \le 1$ and sign preserving for a divergent velocity field when $c \le 1/2$. The anti-diffusive velocity is divergent, so the Courant number based on the anti-diffusive velocity must be less than $1/2$ for the explicit MPDATA to be sign preserving. Smolarkiewicz and Szmelter (2005)

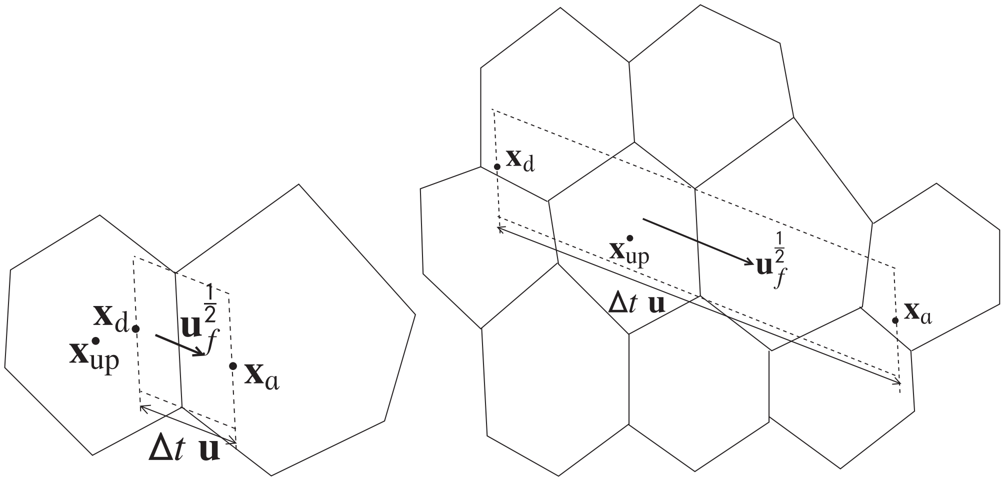


FIGURE 2 The volume that is swept through face f in one time step and the departure and arrival points, $\mathbf{x}_\mathrm{d}$ and $\mathbf{x}_\mathrm{a}$, for small (left) and large (right) Courant numbers

showed that the median dual MPDATA discretisation of the anti-diffusive flux satisfies this criterion for $c \in [0,1]$. In order to guarantee sign preservation in this paper, we limit the anti-diffusive flux so that

$$|V_f/U_f| \le \frac{1}{2}. \tag{14}$$

For the tests presented in Section 3, the anti-diffusive fluxes satisfied Equation (14) without limiting, but we have not (yet) proven if this will always hold.

2.2 | Adaptively implicit MPDATA

The adaptively implicit scheme is a generalisation of Crank–Nicolson with off-centring θ that can vary in space. θ_f is defined on faces (for conservation), so the time stepping is defined as

$$\psi_\mathrm{c}^{n+1} = \psi_\mathrm{c}^{n} - \frac{\Delta t}{\mathcal{V}_\mathrm{c}} \sum_{f\in C} \{(1-\theta_f)\psi_f^n + \theta_f \psi_f^{n+1}\} U_f. \tag{15}$$

This is second order in time only for $\theta_f = \frac{1}{2}$ globally. We will next derive the MPDATA anti-diffusive flux that corrects a scheme that is first-order accurate in space and off-centred by θ in time. So the first step, before the MPDATA correction, is

$$\psi_\mathrm{c}^{1} = \psi_\mathrm{c}^{n} - \frac{\Delta t}{\mathcal{V}_\mathrm{c}} \sum_{f\in C} \left\{(1-\theta_f)\psi_\mathrm{up}^n + \theta_f \psi_\mathrm{up}^{1}\right\} U_f. \tag{16}$$

The proof that this first step gives positive, bounded, and hence stable, solutions for non-divergent velocity fields on arbitrary meshes is provided in Appendix A.

To find the second-order approximation of $\psi_f^{n+1/2}$ for non-divergent flow, we consider a linear combination of ψ at the departure point at t^n and ψ at the arrival point at t^{n+1}:

$$\psi_f^{n+1/2} = (1-\theta_f)\psi_\mathrm{d}^n + \theta_f \psi_\mathrm{a}^{n+1}, \tag{17}$$

where the locations of the departure and arrival points are shown in Figure 2 and are given by

$$\mathbf{x}_\mathrm{d} = \mathbf{x}_f - \frac{\Delta t}{2}\mathbf{u}_f, \tag{18}$$

$$\mathbf{x}_\mathrm{a} = \mathbf{x}_f + \frac{\Delta t}{2}\mathbf{u}_f. \tag{19}$$

The values of ψ at the departure and arrival points can be approximated by

$$\psi_\mathrm{d}^n = \psi_\mathrm{up}^n + \left(\mathbf{x}_\mathrm{d} - \mathbf{x}_\mathrm{up}\right)\cdot\nabla\psi^n, \tag{20}$$

$$\psi_\mathrm{a}^{n+1} = \psi_\mathrm{up}^{n+1} + \left(\mathbf{x}_\mathrm{a} - \mathbf{x}_\mathrm{up}\right)\cdot\nabla\psi^n. \tag{21}$$

Substituting these into Equation (17) gives

$$\begin{aligned}\psi_f^{n+1/2} &= (1-\theta_f)\psi_\mathrm{up}^n + \theta_f\psi_\mathrm{up}^{n+1}\\ &\quad + (\mathbf{x}_f - \mathbf{x}_\mathrm{up})\cdot\nabla\psi^n - (1-2\theta_f)\tfrac{\Delta t}{2}\mathbf{u}\cdot\nabla\psi^n,\end{aligned} \tag{22}$$

where $\mathbf{x}_\mathrm{up}$ is the centre of the cell upwind of face f. This correction is not stable for Courant number $c > 2$ or $\theta > 1/2$ (Appendix B). For stability for all c and second-order accuracy where $\theta \le 1/2$ the correction step is

$$\psi_\mathrm{c}^{n+1} = \psi_\mathrm{c}^{1} + \frac{\Delta t}{\mathcal{V}_\mathrm{c}} \sum_{f\in C} \psi_\mathrm{vup}^{1} V_f, \tag{23}$$

where

$$V_f = \mathbf{v}_f\cdot\mathbf{S}_f = \frac{U_f}{\psi}\left[(\mathbf{x}_f - \mathbf{x}_\mathrm{up})\cdot\nabla\psi^n - \chi\frac{\Delta t}{2}\mathbf{u}\cdot\nabla\psi^n\right] \tag{24}$$

and where

$$\chi = \max(1 - 2\theta_f, 0), \tag{25}$$

where spatial discretisation is as in Section 2.1. Equation (25) gives a first-order error in time for $\theta > 1/2$ which is only used for large Courant numbers ($c > 2$). It is stable on a uniform one-dimensional grid (Appendix B) but on an arbitrary mesh, some smoothing is needed when $\theta > 0$ (Section 2.2.1).

Appendix B shows that the first (diffusive) step of the adaptively implicit MPDATA scheme, Equation (16), is stable and bounded when

$$\theta \geq \max\left(1 - \frac{1}{c}, 0\right), \tag{26}$$

with the Courant number c for an arbitrary mesh defined as in Equation (13). Equation (26) can be used to set θ_f based on the values of the Courant number in the cells either side, c_{up} and c_{down}, with a degree of safety added to avoid reaching the stability limits:

$$\theta_f = \max\left\{1 - \frac{1}{c_{\text{up}} + 0.25}, 1 - \frac{1}{c_{\text{down}} + 0.25}, 0\right\}. \tag{27}$$

2.2.1 | Additional smoothing for large Courant numbers

Appendix B shows that a linearised version of the adaptively implicit MPDATA is unconditionally stable on a uniform, one-dimensional grid. However, this does not carry over onto an arbitrary mesh. Therefore, V_f is smoothed where $\theta > 0$. First, a cell centre anti-diffusive flux is reconstructed from surrounding fluxes:

$$\mathbf{v}_{\text{c}} = \left(\sum_{f\in C}\mathbf{S}_f\mathbf{S}_f^{\text{T}}\right)^{-1}\sum_{f\in C}\mathbf{S}_f V_f, \tag{28}$$

which is the standard reconstruction of vectors from fluxes implemented in OpenFOAM; $\sum_{f\in C}\mathbf{S}_f\mathbf{S}_f^{\text{T}}$ is a tensor that can be inverted and pre-calculated for each cell. This is a second-order accurate, least-squares reconstruction that reconstructs a uniform vector field exactly. The reconstructed velocity is then interpolated back onto faces and the dot product taken with $\mathbf{S}_f$ to get a smoothed flux. The smoothed flux is used for faces with $\theta_f > 0$ and for all faces of a cell if that cell has one face with $\theta_f > 0$:

$$V_f = \begin{cases} V_f \text{ from Equation (24)} & \text{if } \theta_f = 0 \text{ and} \\ & \theta_{f'} = 0 \;\forall\, f' \in (C, N) \text{ of } f \\ \mathbf{v}_{cf} \cdot \mathbf{S}_f & \text{otherwise,} \end{cases} \tag{29}$$

where $\mathbf{v}_{cf}$ is the reconstructed velocity $\mathbf{v}_{\text{c}}$ linearly interpolated from cell centres to faces. The notation $\forall\, f' \in (C, N)$ of f means for all faces f' that are faces of cells C and N, which are the cells surrounding face f.

2.3 | Linear equation solver

The first-order upwind adaptively implicit advection creates a sparse, asymmetric M-matrix M with positive elements on the diagonal and negative off- diagonal elements. To create the matrix equation, Equation (16) is rearranged so that the vector of new ψ^1 values ($\underline{\psi}^1$) is a linear combination of old ψ^n values ($\underline{\psi}^n$):

$$M\underline{\psi}^1 = N\underline{\psi}^n, \tag{30}$$

where

$$M_{ij} = \begin{cases} 1 + \frac{\Delta t}{V_i}\sum_{f\in i}\theta_f \max\{U_f, 0\} & \text{for } i = j \\ -\frac{\Delta t}{V_i}\theta_f \max\{-U_f, 0\} & \text{where } f \text{ is between} \\ & \text{cells } i \text{ and } j \end{cases} \tag{31}$$

and

$$N_{ij} = \begin{cases} 1 - \frac{\Delta t}{V_i}\sum_{f\in i}(1 - \theta_f)\max\{U_f, 0\} & \text{for } i = j \\ +\frac{\Delta t}{V_i}(1 - \theta_f)\max\{-U_f, 0\} & \text{where } f \text{ is between} \\ & \text{cells } i \text{ and } j. \end{cases} \tag{32}$$

Matrix N is, of course, not created because the right-hand side vector entries can be evaluated directly. If the flow is non-divergent, then $\sum_{f\in i} U_f = 0$, which implies that M is strictly diagonally dominant. Note that M has no off-diagonal elements where the time stepping is explicit. M will not be diagonally dominant at row i if the volume flux into cell i in one time step is greater than the volume flux out in that time step plus the cell volume. This situation is not likely for atmospheric modelling as the atmosphere is low Mach number, but it would require either a smaller time step or a matrix solver suitable for non-diagonally dominant matrices.

The resulting linear equation system is solved with the standard OpenFOAM bi-conjugate gradient solver with a diagonal incomplete lower–upper preconditioner. Solver tolerance and iteration counts are discussed in Section 3.4.

2.4 | Infinite-gauge MPDATA

Smolarkiewicz and co-workers (Smolarkiewicz and Clark, 1986; Smolarkiewicz and Grabowski, 1990) introduce the idea of adding a constant (or gauge) to the quantity transported by MPDATA and removing the gauge after it has been transported. This has the effect of moving the transported quantity away from zero, thus moving the zero bound on the transported variable; the transported quantity is no longer guaranteed to remain positive. The scheme can be reformulated assuming the gauge is infinite (Smolarkiewicz and Margolin, 1998; Smolarkiewicz, 2006), which can improve accuracy and enables transport of signed quantities, such as velocity. This infinite-gauge variant of MPDATA is a realisation of the Lax–Wendroff scheme, and can be used with the adaptively implicit time stepping exactly as it is used with standard, explicit MPDATA. As with the explicit MPDATA, monotone solutions can be achieved using FCT, as described next.

2.5 | FluxCorrected Transport (FCT) with Implicit Time Stepping

Zalesak (1979) state that FCT can be used with implicit time stepping, although we have not found examples of this in the literature. In fact, the algorithm as described by Zalesak (1979) does not guarantee monotonicity when used with implicit time stepping. This is because Zalesak (1979) bound the tracer at t^{n+1} by the diffusively transported tracer at t^{n+1} and the tracer at t^n at the current and upwind grid points. The tracer at t^n at the current and upwind grid points will not have suitable bounds if the tracer can move a long distance in one time step. When using implicit time stepping and large Courant numbers, local extrema can be advected by more than one mesh cell in one time step, so local bounds from the previous time step no longer apply. We therefore define two variants of FCT to work with implicit time stepping. One guarantees monotonicity (as defined in Appendix A) and the other guarantees global boundedness given user-defined bounds.

The first step of FCT is to advect using a monotonic, diffusive scheme to calculate ψ^1. Appendix A shows that the first-order upwind in space, adaptively implicit in time scheme, Equation (16), provides this solution for arbitrary Courant numbers. The next step is to calculate the allowable minima and maxima for each cell, which we will call $\psi_{\min}$ and $\psi_{\max}$. If we seek boundedness within pre-defined bounds then $\psi_{\min}$ and $\psi_{\max}$ are these bounds. Otherwise $\psi_{\min}$ and $\psi_{\max}$ are the local extrema of ψ^1 in the current and neighbouring cells. Explicit FCT also uses ψ^n, which widens the bounds. Consequently, FCT for implicit, monotonic advection will be more diffusive because of the use solely of ψ^1 to define the local bounds:

$$\text{for cell } C \ \psi_{\min} = \min_{N\in C}\{\psi_N^1\}, \tag{33}$$

$$\text{for cell } C \ \psi_{\max} = \max_{N\in C}\{\psi_N^1\}, \tag{34}$$

where in both cases N are the face neighbours of C. We next define the maximum allowable amount that each cell can rise or fall by and use the same notation as Zalesak (1979):

$$Q_p = \psi_{\max} - \psi^1, \tag{35}$$

$$Q_m = \psi^1 - \psi_{\min}. \tag{36}$$

We next need to modify the non-monotonic MPDATA high-order flux corrections (HOCs). The HOC is the MPDATA flux correction V_f from Equation (29) multiplied by ψ^1 at the upwind cell (upwind defined relative to V_f):

$$F_{f\text{HOC}} = \psi_{\text{up}}^1 V_f. \tag{37}$$

From this we calculate the total high-order flux that enters (P_p) and leaves (P_m) each cell:

$$P_p = -\frac{\Delta t}{\mathcal{V}_{\text{c}}} \sum_{f\in C} \min\{F_{f\text{HOC}}, 0\}, \tag{38}$$

$$P_m = \frac{\Delta t}{\mathcal{V}_{\text{c}}} \sum_{f\in C} \max\{F_{f\text{HOC}}, 0\}. \tag{39}$$

Next, we find the ratios of the allowable total fluxes to the actual high-order fluxes:

$$R_p = \begin{cases} \min\left\{1, \frac{Q_p}{P_p}\right\} & \text{if } P_p > 0 \\ 0 & \text{otherwise,} \end{cases} \tag{40}$$

$$R_m = \begin{cases} \min\left\{1, \frac{Q_m}{P_m}\right\} & \text{if } P_m > 0 \\ 0 & \text{otherwise.} \end{cases} \tag{41}$$

Finally, we find the coefficient to multiply $F_{f\text{HOC}}$ in order to achieve either a monotonic solution or a solution with the required bounds:

$$F_f = \begin{cases} F_{f\text{HOC}} \min\{R_{pN}, R_{mC}\} & \text{if } F_{f\text{HOC}} \geq 0 \\ F_{f\text{HOC}} \min\{R_{pC}, R_{mN}\} & \text{otherwise,} \end{cases} \tag{42}$$

where

$$(\mathbf{x}_N - \mathbf{x}_C) \cdot \mathbf{S}_f > 0 \tag{43}$$

and cells C and N are either side of face f. Then, the final update is

$$\psi_c^{n+1} = \psi_c^1 - \frac{\Delta t}{\mathcal{V}_c} \sum_{f \in C} F_f, \tag{44}$$

which is monotonic if Equations 33 and 34 are used as bounds on ψ. Alternatively, global bounds such as $[\psi_{\min}, \psi_{\max}] = [0, 1]$ can be specified.

3 | ADVECTION TEST CASES

3.1 | One-dimensional advection

The first test of the adaptively implicit MPDATA is one-dimensional with uniform velocity. Variable resolution is used so that the Courant number varies in space, and implicit time stepping is used only where resolution is fine. The variable-resolution grids have resolution a factor of R finer in the middle of the unit length domain than the end points. There are n cells (n unique grid points) in the unit length and a constant ratio $r = R^{2/(n-2)}$ between successive cells in the first half of the domain and $1/r$ in the second half. Therefore, the resolution of cell i is

$$\Delta x_i = \begin{cases} \frac{1}{2} R r^{-i} \frac{1-r}{1-rR} & i \le \frac{n}{2} - 1 \\ \frac{1}{2} R r^{(n/2)-i} \frac{1-r}{1-rR} & i \ge \frac{n}{2}. \end{cases} \tag{45}$$

We use smooth initial conditions for evaluating convergence with resolution and mixed initial conditions for inspecting boundedness and overall quality of solution:

$$\psi^0_{\text{smooth}} = \begin{cases} \frac{1}{2}[1 + \cos \pi(4x - 1)] & x \in [0, 0.5] \\ 0 & \text{otherwise,} \end{cases} \tag{46}$$

$$\psi^0_{\text{mixed}} = \begin{cases} \frac{1}{2}[1 + \cos \pi(4x - 1)] & x \in [0, 0.5] \\ 1 & x \in [0.6, 0.8] \\ 0 & \text{otherwise.} \end{cases} \tag{47}$$

All simulations use a velocity of $u = 1$ and run for one time unit so that the tracer travels one complete revolution around the periodic domain.

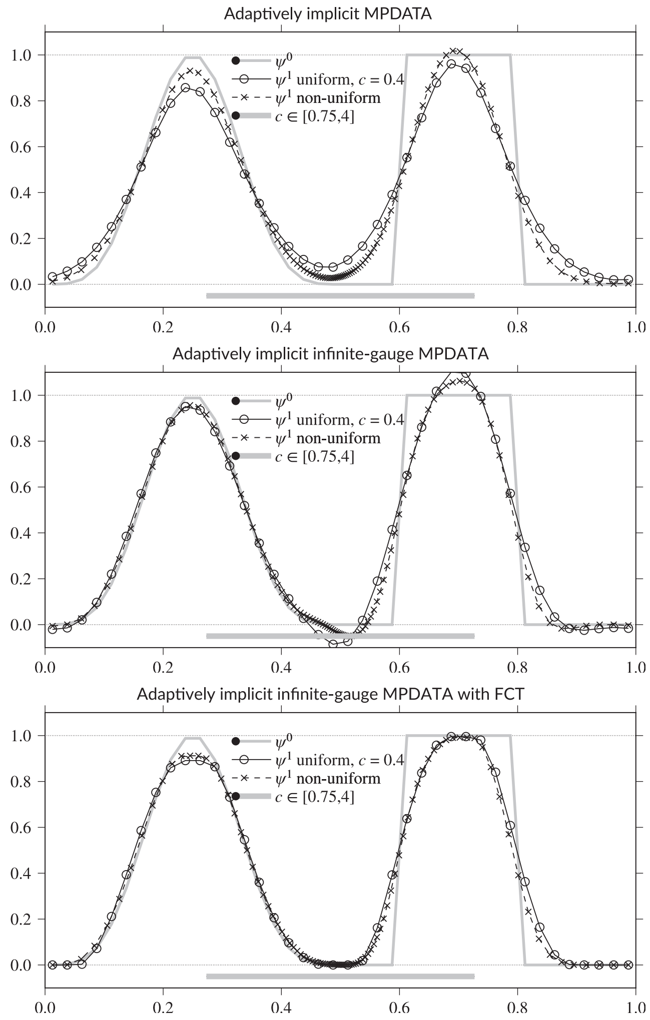


FIGURE 3 Advection once around a periodic domain starting from mixed initial conditions, ψ^0, using 40 grid points for the uniform resolution and 100 grid points for the resolution with a factor $R = 10$ between finest and coarsest. The regions where the non-uniform resolution has a Courant number greater than 0.75 (where implicit time stepping is used) are shaded grey

Figure 3 shows solutions starting from the mixed initial conditions using 100 time steps each of length $\Delta t = 0.01$. The uniform resolution has 40 cells giving a uniform Courant number of 0.4 (meaning that the time stepping is purely explicit). The non-uniform resolution has 100 cells with $R = 10$, giving a Courant number in the range $c \in [0.4, 4]$ so that implicit time stepping is used where $c > 0.75$. As expected, the MPDATA results (top row of Figure 3) are always positive for both the uniform resolution (explicit time stepping) and the non-uniform resolution (adaptively implicit). The non-uniform resolution produces a stable overshoot above the square wave, which can happen with MPDATA without limited fluxes. The infinite-gauge version (middle row of Figure 3) produces undershoots and overshoots, and the solution is more accurate in the region of the smooth wave. Neither the uniform (explicit) or non-uniform (adaptively implicit) results appear more accurate than the other, although more grid points are used for the variable resolution. The infinite-gauge results using FCT (bottom row of Figure 3)

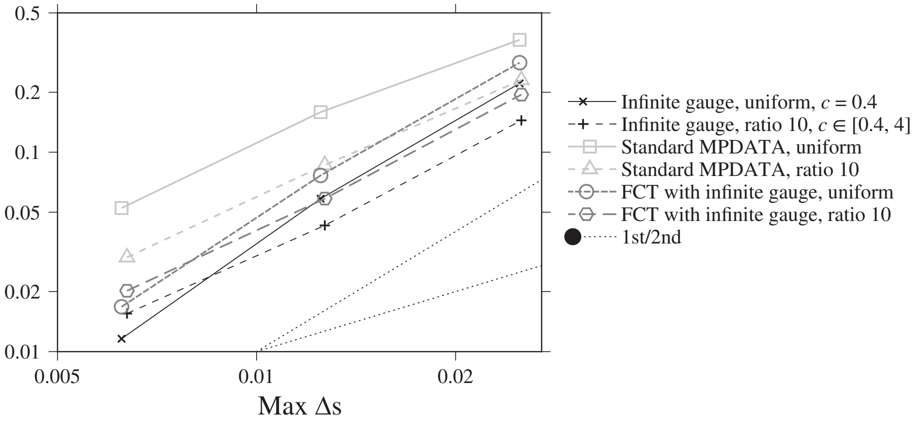
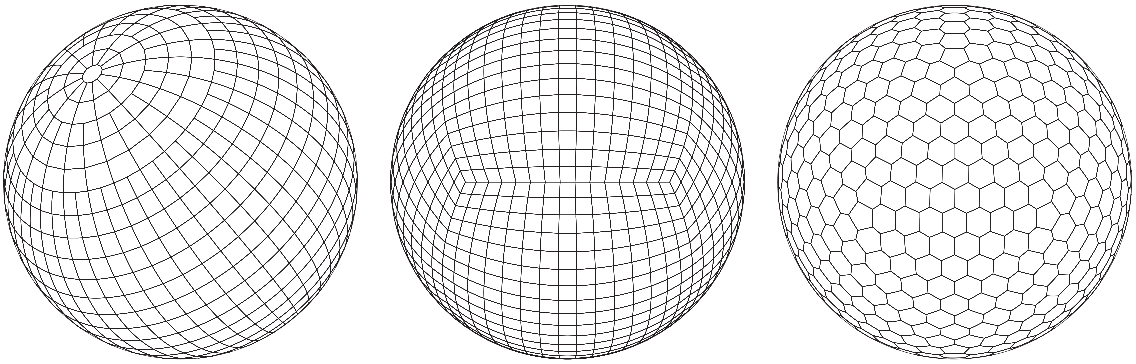


FIGURE 4 Convergence of the ℓ_2 error norm with resolution of the one-dimensional advection of the smooth initial conditions ψ^0. The uniform resolutions use 20, 40, and 80 grid points, and the non-uniform resolutions use 50, 100, and 200 grid points with ratio $R = 10$. Both use time steps of $\Delta t = 0.02$, 0.01, and 0.005. Dotted lines show the slope of first- and second-order convergence. FCT: flux-corrected transport



Skipped latitude-longitude
48×24, 818 cells

Gnomonic cubed-sphere
$15 \times 15 \times 6 = 1{,}350$ cells

Hexagonal icosahedron
642 cells

FIGURE 5 Some common meshes of the sphere viewed from above a point at a latitude of 45°

are bounded, demonstrating the correct application of the flux corrections applied to adaptively implicit time stepping. The uniform (explicit) or non-uniform (adaptively implicit) results appear similar.

Convergence with resolution for all schemes on uniform and non-uniform meshes starting from the smooth initial conditions is shown in Figure 4. The time step is scaled with the resolution so that the uniform meshes retain $c = 0.4$ at all resolutions and the non-uniform meshes retain $c \in [0.4, 4]$. The standard and infinite-gauge MPDATA with and without FCT give second-order convergence. Even though the non-uniform grid means that the Courant number reaches 4 at the centre of the domain, the convergence remains strong.

As with explicit MPDATA (Smolarkiewicz and Grabowski, 1990; Smolarkiewicz, 2006), the adaptively implicit infinite-gauge version is less dissipative than the version without a gauge. Again, as with explicit MPDATA, the gauge version remains more accurate with the application of FCT. However, FCT removes the symmetry of the diffusion of the step wave and tends to distort rather than diffuse the smooth initial conditions.

3.2 | Spherical meshes

Advection test cases using adaptively implicit MPDATA calculated using various meshes are presented. There is no clearly optimal mesh of the sphere for atmospheric modelling (example meshes in Figure 5). Numerical methods need to be designed to allow for one or more of the following features of meshes of the sphere:

1. Latitude–longitude meshes are orthogonal and have uniform resolution following coordinate lines, but they have severe convergence of mesh lines towards two poles; so, numerical methods are needed that can cope with very large Courant numbers. We use a latitude–longitude mesh with a cell at each pole.
2. Hexagonal and triangular meshes of the sphere are quasi-uniform, but they cannot be all three of:
    (a) orthogonal (mesh lines and cell centre to cell centre lines cross at right angles);
    (b) centroidal (cell centres are at cell centroids);

   (c) non-skew (cell centre to cell centre lines bisect mesh lines).

   This means that special numerical treatment is needed in order to achieve second-order accuracy.

3. Quasi-uniform versions of the cubed sphere are non-orthogonal with large distortions (skewness) at cube edges and corners, so numerical methods are needed that maintain accuracy at these distortions. The cubed sphere in Figure 5 uses the gnomonic projection (Rančić *et al.*, 1996).
4. Skipped latitude–longitude meshes have factor-of-two reductions in resolution in the longitudinal direction at a few latitudes to prevent the mesh lines converging. At latitudes where the resolution reduces, the meshes can be treated as non-conforming so that two quadrilateral cells are connected to one edge of the adjacent quadrilateral cell, or conforming with two quadrilaterals connected to adjacent, aligned edges of a distorted pentagon. The implementation described here treats them as conforming.

All of the meshes were decomposed into four domains for parallel processing with MPI.

3.3 | Deformational flow

Lauritzen *et al.* (2012) describe deformational flow test cases to demonstrate a number of numerical properties of an advection scheme including order of convergence and monotonicity. We are using the non-divergent wind field, which deforms and translates the initial conditions so that the final solution ($t = T = 5$) should be identical to the initial conditions ($t = 0$). The wind is defined by a stream function Ψ based on latitude φ, longitude λ, time t, and the radius of the sphere $R = 1$:

$$\Psi(\lambda,\varphi,t) = \frac{10R}{T}\sin^2\left(\lambda - \frac{2\pi t}{T}\right)\cos^2\varphi\cos\frac{\pi t}{T} - \frac{2\pi R}{T}\sin\varphi. \tag{48}$$

3.3.1 | Gaussian hills

The Gaussian hills initial conditions are smooth and so can be used to measure the numerical order of convergence. The initial conditions of the tracer ψ_0 are given in terms of the three-dimensional position vector $\mathbf{x}$ in Cartesian coordinates:

$$\psi_0(\mathbf{x}) = 0.95\{\exp[-5(\mathbf{x}-\mathbf{x}_1)^2] + \exp[-5(\mathbf{x}-\mathbf{x}_2)^2]\}, \tag{49}$$

where

$$\mathbf{x}_i = (R\cos\varphi_i\cos\lambda_i, R\cos\varphi_i\sin\lambda_i, R\sin\varphi_i), \tag{50}$$

$$(\lambda_1,\varphi_1) = (5\pi/6, 0), \tag{51}$$

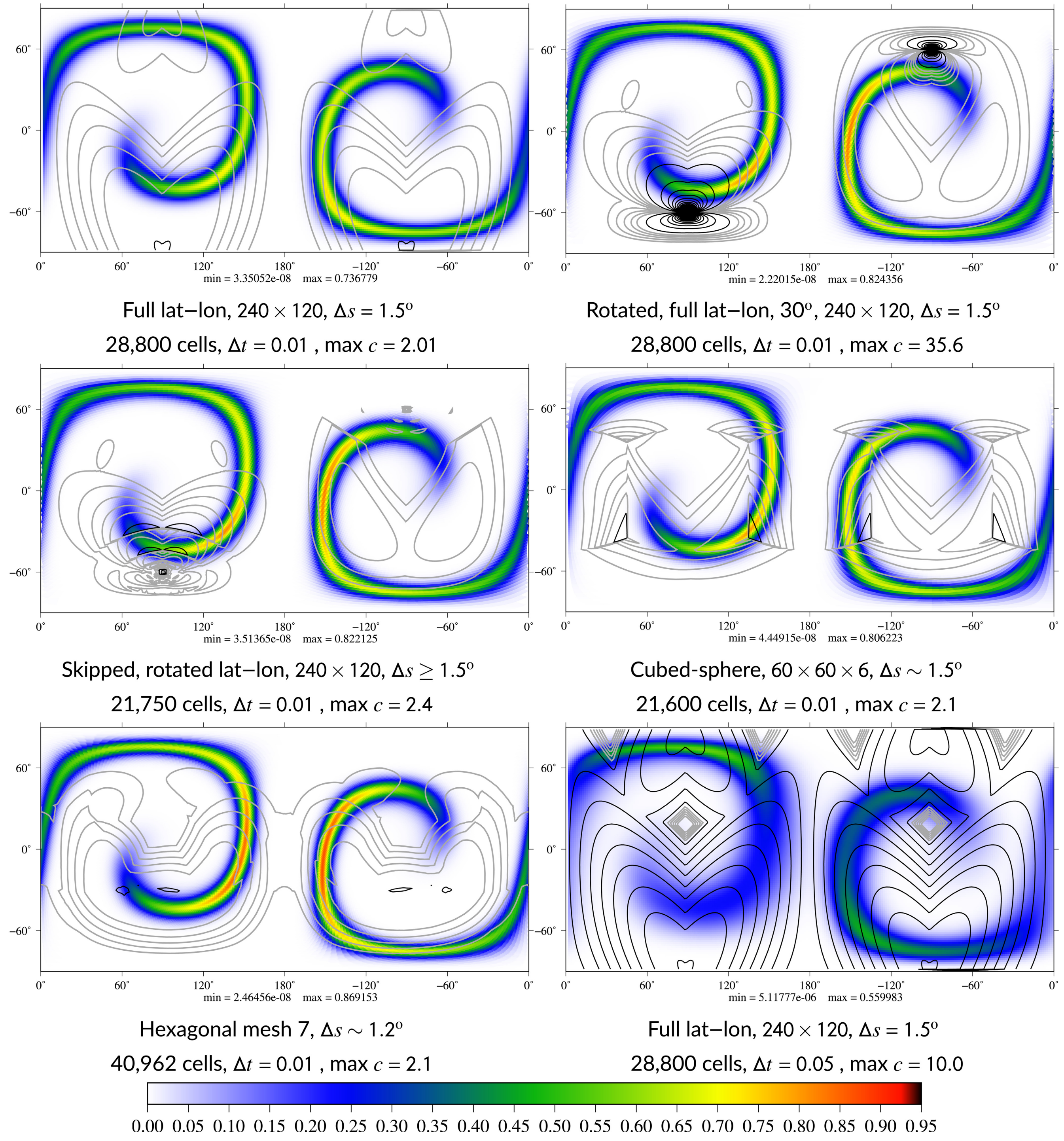
$$(\lambda_2,\varphi_2) = (7\pi/6, 0). \tag{52}$$

The tracer concentrations at $t = 2.5$ are shown in Figure 6 calculated on four different meshes of the sphere and for a 30° rotated version of the latitude–longitude mesh, all at a similar resolution. These use the standard adaptively implicit MPDATA without FCT.

Simulations using all the meshes in Figure 6 use a time step of 0.01 (500 time steps in total), giving a Courant number of around 2, so that the simulations would be unstable if a purely explicit scheme were used. Results on the full latitude–longitude mesh with a time step of 0.05 are also shown, leading to a maximum Courant number of around 10. Courant numbers at $t = 0$ are contoured in Figure 6. Spatial resolutions and time steps are shown in Table 1.

The flow goes to zero at the North and South Poles, so the convergence of meridians of the unrotated latitude–longitude mesh does not lead to large Courant numbers. However, when the mesh is rotated by 30°, high winds cross the poles of the mesh so the maximum Courant number goes up to 70 (the contours in Figure 6 show the Courant number at $t = 0$). These large Courant numbers do not lead to instability, a lack of sign preservation, or visible artefacts in the solution. The largest Courant numbers are removed on the rotated, skipped latitude–longitude mesh, although Courant numbers >2 are present at $t = 0$, just poleward of the change in longitudinal resolution. On the cubed sphere, the Courant number is largest near the cube corners due to mesh distortions and smaller cells. Some mesh imprinting is visible along the cube edges, although this does not lead to a lack of sign preservation. The hexagonal icosahedral meshes are the most uniform meshes of the sphere and so there are no sharp spikes in the Courant number. The results from the hexagonal mesh appear accurate, but note that this mesh has higher resolution than the other meshes. The solutions using a larger time step on a full latitude–longitude mesh have severely degraded accuracy, with the Courant number being large over most of the domain and so very little of the high-order MPDATA correction can be applied.

MPDATA is, by design, sign preserving but not monotonic. The adaptively implicit MPDATA retains this feature on all of the meshes tested and displayed in Figure 6. The minimum and maximum tracer values for all time steps for each of the meshes in Figure 6 are shown at the top of Figure 7. All of the minima remain positive and very close to zero. The maxima decrease due to numerical diffusion but do not decrease monotonically, as expected using the standard MPDATA.



Full lat–lon, 240 × 120, $\Delta s = 1.5^o$
28,800 cells, $\Delta t = 0.01$, max $c = 2.01$

Rotated, full lat–lon, 30^o, 240 × 120, $\Delta s = 1.5^o$
28,800 cells, $\Delta t = 0.01$, max $c = 35.6$

Skipped, rotated lat–lon, 240 × 120, $\Delta s \geq 1.5^o$
21,750 cells, $\Delta t = 0.01$, max $c = 2.4$

Cubed-sphere, 60 × 60 × 6, $\Delta s \sim 1.5^o$
21,600 cells, $\Delta t = 0.01$, max $c = 2.1$

Hexagonal mesh 7, $\Delta s \sim 1.2^o$
40,962 cells, $\Delta t = 0.01$, max $c = 2.1$

Full lat–lon, 240 × 120, $\Delta s = 1.5^o$
28,800 cells, $\Delta t = 0.05$, max $c = 10.0$

FIGURE 6 Deformational flow on the sphere. The shading show the tracer at $t = 2.5$ (piecewise constant in each cell). The grey contours show the Courant number at $t = 0$ from 0.8 to 1.8 every 0.2, and the black contours are from 2 to 50 every 1 [Colour figure can be viewed at wileyonlinelibrary.com]

The maximum and mean Courant numbers for each time step for each of the meshes in Figure 6 are shown in the middle row of Figure 7. The maximum Courant number for all meshes is >1 using the time step of 0.01 and is minimum at the middle of the simulation ($t = 2.5$). The maximum Courant number for the rotated latitude–longitude mesh reaches 70 and is always much larger than 1, which does not appear to significantly reduce the accuracy. The mean Courant numbers (dashed) are below or close to 1 throughout, which helps to maintain accuracy apart from for the simulation with a larger time step.

TABLE 1 Resolutions and time steps for deformational advection

Mesh type	Nominal resolution	No. cells	Δs°	Δt	Figure
Latitude–longitude	120×60	7,080	3.0	0.02	7c, 8b
	240×120	28,800	1.5	0.01	6,7,8
	480×240	114,720	0.75	0.005	7c, 8b, 9
Skipped latitude–longitude	48×24	864	7.5		5
	120×60	5,310	3.0	0.02	7c, 8b
	240×120	21,750	1.5	0.01	6,7,8
	480×240	88,470	0.75	0.005	7c, 8b, 9
Cubed sphere	$15 \times 15 \times 6$	1,350	6.4		5
	$30 \times 30 \times 6$	5,400	3.2	0.02	7c, 8b
	$60 \times 60 \times 6$	21,600	1.6	0.01	6,7,8
	$120 \times 120 \times 6$	86,400	0.8	0.005	7c, 8b, 9
Hexagonal–icosahedral	HR4	642	9.5		5
	HR6	10,242	2.4	0.02	7c, 8b
	HR7	40,962	1.2	0.01	6,7,8
	HR8	163,842	0.6	0.005	7c, 8b, 9

Note: Δs is a typical cell centre to cell centre distance in degrees latitude.

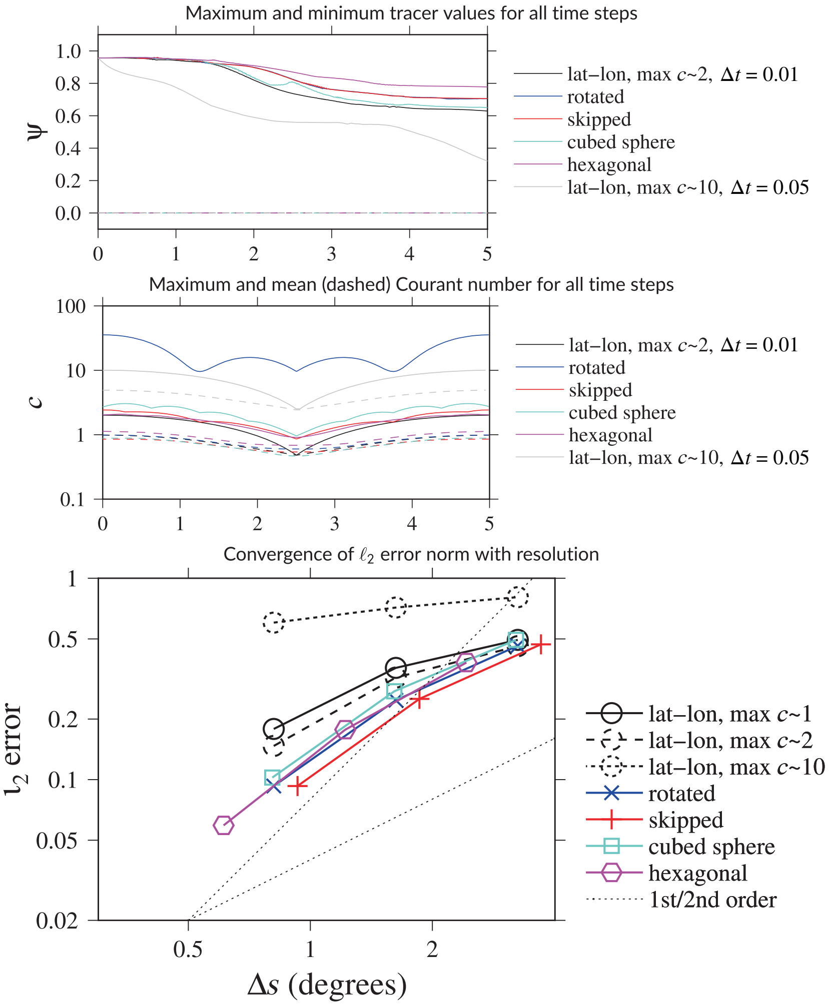


FIGURE 7 Diagnostics of the results for the deformational flow of the Gaussian hills with standard adaptively implicit MPDATA without flux-corrected transport. Top and middle are diagnostics of the simulations shown in Figure 6. Bottom includes other resolutions. Mesh and time-step details in Table 1

The convergence of the ℓ_2 error norm with resolution is shown in the bottom row of Figure 7. The mesh resolutions and time steps for these simulations are given in Table 1. The resolution to time-step ratio is kept constant along each line. Included in this graph are simulations using half the time step and five times the time step for the latitude–longitude mesh (giving maximum Courant numbers around 1 and around 10) in order show the impact of varying the mean Courant number. Reducing the time step to get $c < 1$ means that the standard explicit MPDATA is used almost everywhere. This increases the error slightly, which can be expected given the smooth flow. When the Courant number is close to 2, θ is close to $1/2$ and the temporal error correction is small. This implies that using second-order adaptively implicit time stepping is more accurate than using first-order time stepping with a correction. However, the adaptively implicit time stepping requires a matrix inversion, and so is more expensive. The simulation with $c \leq 10$ is much less accurate because the temporal correction is not applied for $c \geq 2$. However, the simulation is still stable and sign preserving.

The convergence with resolution in Figure 7 is around first order at coarse resolution and approaches second order at higher resolution, as expected for a second-order scheme (the asymptotic convergence is second order). The errors in Figure 7 are similar to the second-order schemes presented in Lauritzen *et al.* (2014).

Better accuracy at the expense of sign preservation can be achieved with the infinite-gauge variant of MPDATA (a realisation of Lax–Wendroff), which works for the adaptively implicit version in the same way as the standard MPDATA (Smolarkiewicz and Clark, 1986). The maximum and minimum values of the tracer for infinite-gauge simulations with the same resolution as those shown in Figure 6 are shown in Figure 8. In comparison with the standard MPDATA simulations (Figure 7), the infinite-gauge results have a smaller reduction in the maximum (because the results are more accurate and hence less diffusive) but the minima are less than zero (spurious undershoots are generated). ℓ_2 errors with resolution are shown at the bottom of Figure 8. The mesh spacing and time steps are the same as in Figure 7 and are

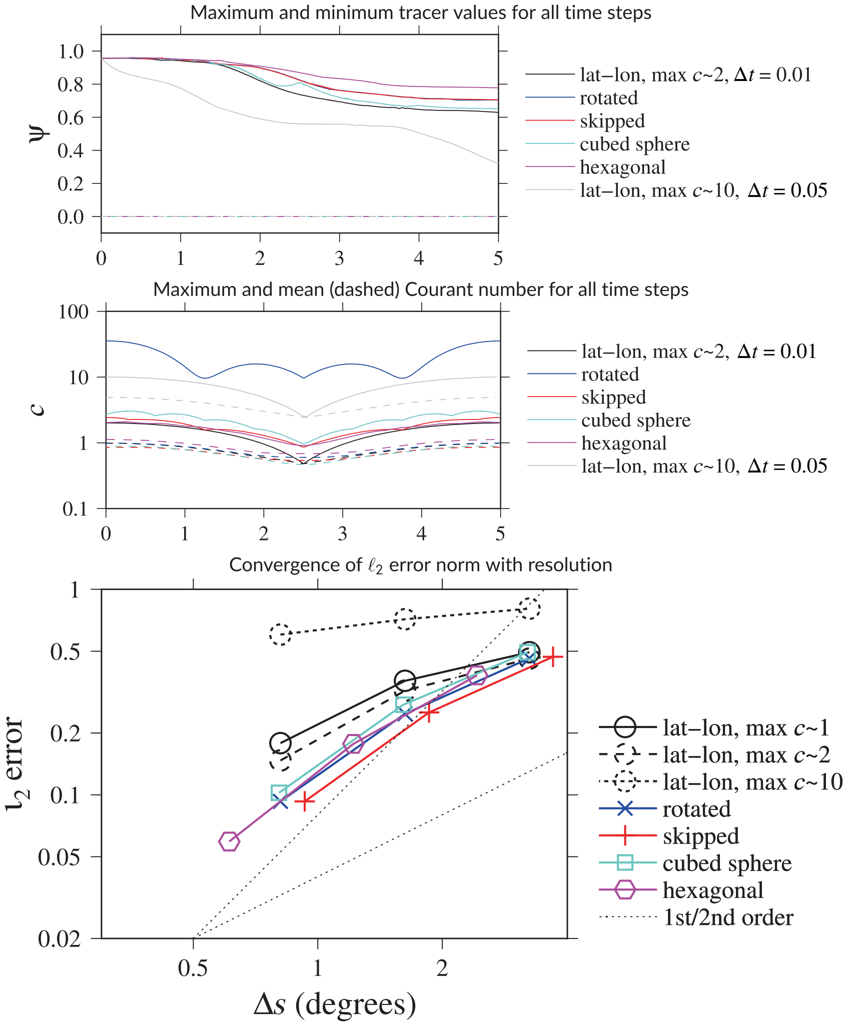


FIGURE 8 Diagnostics of the results for the deformational flow of the Gaussian hills with adaptively implicit infinite gauge MPDATA without flux-corrected transport. Other settings are the same as Figure 7

shown in Table 1. The order of convergence is higher and the ℓ_2 errors lower than standard MPDATA results (Figure 7).

3.3.2 | Slotted cylinders

Deformational advection of slotted cylinders tests the implementation of limiters. Lauritzen *et al.* (2012) recommend the same deformational velocity field as for the Gaussian hills with initial tracers defined by

$$\psi_0(\lambda,\phi)=\begin{cases}1 & \text{if } r_i\le r \text{ and } |\lambda-\lambda_i|\ge \frac{r}{6R} \text{ for } i=1,2\\ 1 & \text{if } r_1\le r \text{ and } |\lambda-\lambda_1|<\frac{r}{6R} \text{ and}\\ & \quad \phi-\phi_1<-\frac{5}{12}\frac{r}{R}\\ 1 & \text{if } r_2\le r \text{ and } |\lambda-\lambda_2|<\frac{r}{6R} \text{ and}\\ & \quad \phi-\phi_2>\frac{5}{12}\frac{r}{R},\\ 0.1 & \text{otherwise}\end{cases} \tag{53}$$

where

$$\mathbf{x}=(R\cos\phi\cos\lambda, R\cos\phi\sin\lambda, R\sin\phi) \tag{54}$$

$$r=R/2, r_i=|\mathbf{x}-\mathbf{x}_i| \tag{55}$$

$$(\lambda_1,\varphi_1)=(5\pi/6,0) \tag{56}$$

$$(\lambda_2,\varphi_2)=(7\pi/6,0). \tag{57}$$

The tracer fields at the end of the simulations ($t=T=5$) are shown in Figure 9 for all meshes at the highest resolution used and at time steps giving Courant numbers of around 2 (see Table 1). This uses the adaptively implicit infinite-gauge MPDATA with FCT (limited to ensure monotonicity). Figure 9 shows that the bounds of the initial conditions are maintained and no new extrema are generated, even on the rotated latitude–longitude mesh where the Courant number reaches 140. The skipped latitude–longitude mesh has sharp jumps in the Courant number (shown at $t=5$) that do not cause artefacts in the solution. This is, to our knowledge, the first monotonic and conservative solution of the advection equation using such a large Courant number.

A simulation using the much larger time step that give a maximum Courant number of 10 on the unrotated latitude–longitude mesh is also shown in Figure 9. Monotonicity is preserved but the solution loses accuracy at this globally large Courant number as the MPDATA correction cannot be applied in full for $c>2$.

3.4 | Solver performance

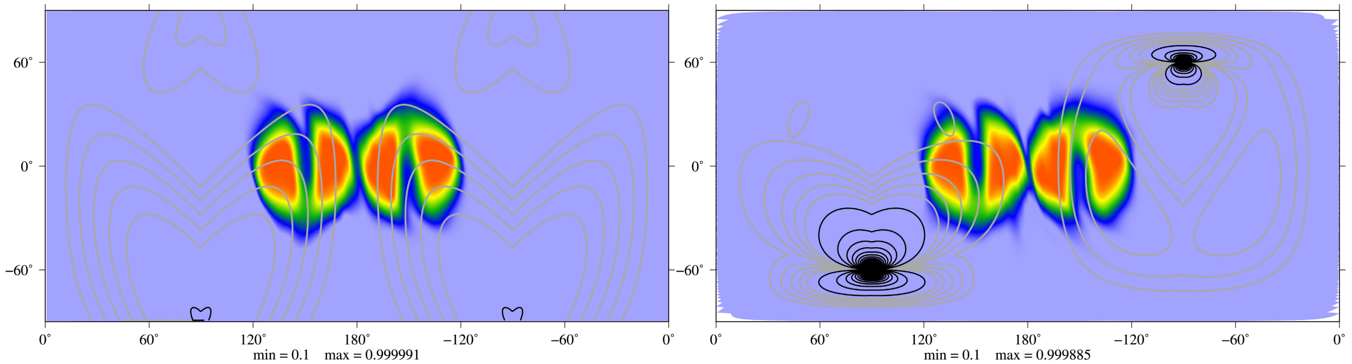
Solver performance is reported for a selection of simulations using full latitude–longitude meshes, as the large inhomogeneity of cell size and large range of Courant numbers could lead to an ill-conditioned matrix and poor solver performance (Tumolo and Bonaventura, 2015). These are compared with solver performance on the highest resolution hexagonal mesh. Each time step consists of one implicit solve using the standard OpenFOAM bi-conjugate gradient solver with a diagonal-based incomplete lower–upper preconditioner. The solver tolerance is

$$\frac{\sum \mathcal{V}_c|\mathbf{y}-A\mathbf{x}|}{\sum \mathcal{V}_c(|\mathbf{y}|-|A\mathbf{x}|)}, \tag{58}$$

for matrix equation $A\mathbf{x}=\mathbf{y}$, where the sum is over all cells of the mesh and $\mathcal{V}_c$ is the cell volume. A tolerance of 10^{-6} is used for all simulations. The first guess of the solver is the state at the previous time step, so the initial residual is small for small time steps.

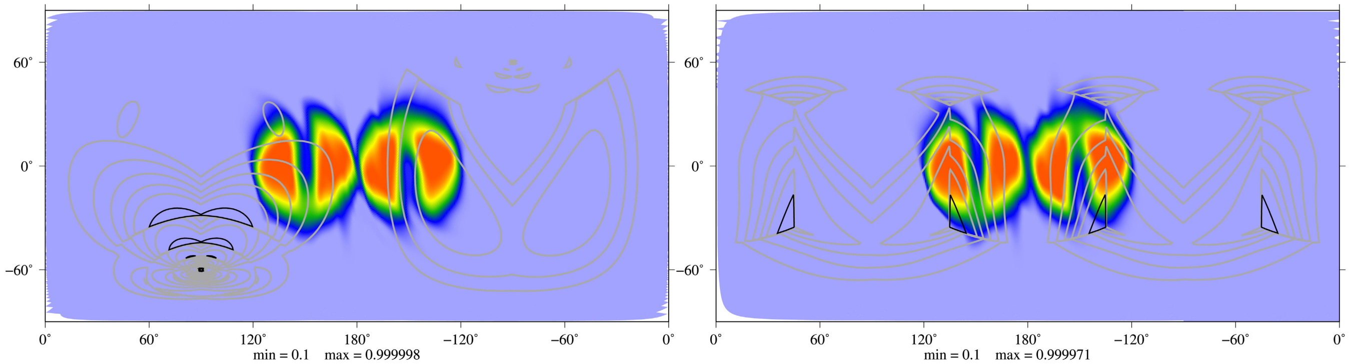
The number of iterations of the solver per time step is shown in Figure 10 for various resolutions and various time steps, both rotated and unrotated, on the latitude–longitude mesh and on the hexagonal mesh. The number of iterations is smallest around time 2.5, when the wind speed is lowest and so the Courant number is smallest. For simulations with the maximum Courant number less than 0.75 in the middle of the simulations, the number of solver iterations drops to zero because the simulation is purely explicit. The simulations represented by black and grey lines have a maximum Courant number of 2, and so the number of iterations is small throughout the simulation. The simulations where the maximum Courant number reaches 10 (in blue) use more iterations, but for the latitude–longitude mesh the number of iterations increases slower than linearly with Courant number, which is necessary for efficiency. However, the hexagonal mesh with a maximum Courant number of 10 uses more than five times as many solver iterations as with a maximum Courant number of 2. This could be because the matrix solver is unsuitable for the reduced sparsity of the hexagonal mesh. The hexagonal mesh has nearly uniform global resolution, and so the Courant number is high globally. This set-up also leads to low accuracy, demonstrating the futility of using implicit methods to achieve large time steps if the Courant number is high everywhere.

The rotated latitude–longitude meshes have very high maximum Courant numbers, but only in limited regions. The residual is a volume average over the whole mesh, so the higher errors near the mesh poles do not prevent global convergence but are still being solved accurately



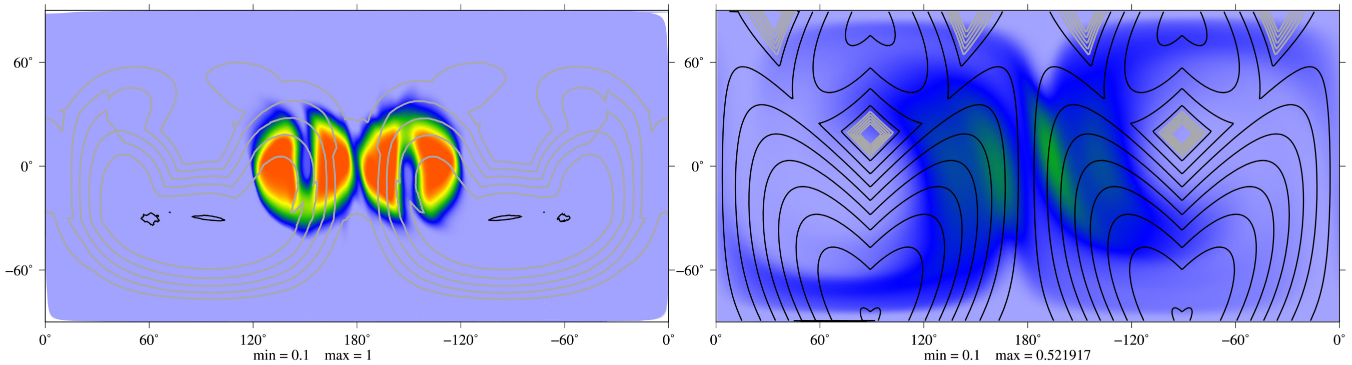
Full lat–lon, 480 × 240, $\Delta s = 0.75^{\mathrm{o}}$
115,200 cells, $\Delta t = 0.005$, max $c = 2.01$

Rotated 30°, full lat–lon, 480 × 240, $\Delta s = 0.75^{\mathrm{o}}$
115,200 cells, $\Delta t = 0.005$, max $c = 70.0$



Skipped, rotated lat–lon, 480 × 240, $\Delta s \geq 0.75^{\mathrm{o}}$
88,470 cells, $\Delta t = 0.005$, max $c = 2.4$

Cubed-sphere, 120 × 120 × 6, $\Delta s \sim 0.75^{\mathrm{o}}$
86,400 cells, $\Delta t = 0.005$, max $c = 3.1$



Hexagonal mesh 8, $\Delta s \sim 0.6^{\mathrm{o}}$
163,842 cells, $\Delta t = 0.005$, max $c = 2.1$

Full lat–lon, 480 × 240, $\Delta s = 0.75^{\mathrm{o}}$
115,200 cells, $\Delta t = 0.025$, max $c = 10.0$

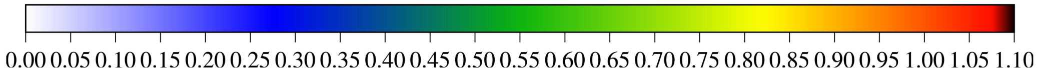


FIGURE 9 Deformational flow on the sphere after five time units. The shading show the piecewise uniform value of the tracer value in each cell. The grey contours show the Courant number from 0.8 to 1.8 every 0.2, and the black contours are from 2 to 50 every 1 [Colour figure can be viewed at wileyonlinelibrary.com]

enough to prevent instability around the pole. Therefore, the rotated mesh simulations do not have high iteration counts for any of the resolutions tested and the accuracy is high. It should be noted that implicit solutions in limited regions will lead to load-balancing problems. Therefore, if it is known in advance that some regions are more likely to need implicit solves; then, smaller domains could be used there.

It should also be noted that higher spatial resolution has little influence on the number of iterations per time

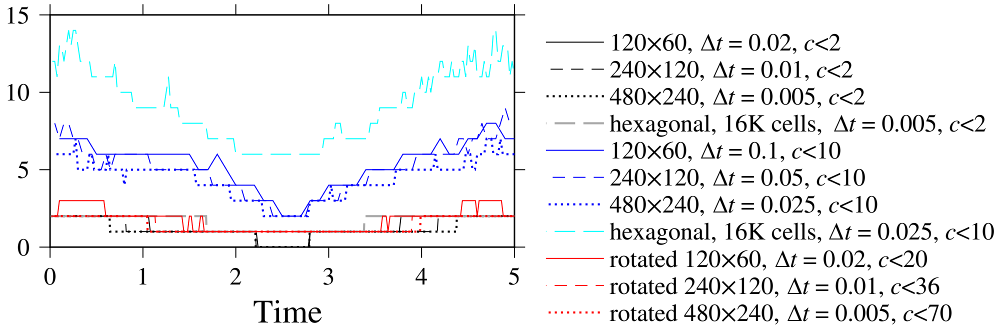


FIGURE 10 Number of solver iterations per time step for simulations on full latitude–longitude meshes and the highest resolution hexagonal mesh [Colour figure can be viewed at wileyonlinelibrary.com]

step for all mesh types and Courant numbers, which is encouraging.

4 | SUMMARY AND CONCLUSIONS

This paper has shown how MPDATA can be extended for adaptively implicit time stepping, enabling Courant numbers much larger than 1. Two-dimensional deformational flow advection test cases on the sphere show that solutions are accurate with Courant numbers >1 over a large fraction of the domain and accurate with local Courant number spikes over 100, such as happens over the pole of a latitude–longitude mesh. There are a number of novel aspects to the paper and the advection scheme presented:

1. An adaptively implicit version of finite-volume MPDATA that is stable for arbitrary Courant numbers and on arbitrary meshes.
2. Transport over the poles of a latitude–longitude mesh at high wind speed without reductions in accuracy or increased cost.
3. Monotonicity for all Courant numbers by adapting FCT (Zalesak, 1979) for implicit time stepping.
4. Accuracy of the adaptively implicit scheme for modest Courant numbers (up to 2) and first-order accuracy as Courant numbers grow beyond 2.
5. A proof that the adaptively implicit time stepping, in combination with first-order upwind spatial discretisation, is bounded.
6. A demonstration of the advection scheme on a variety of meshes of the sphere.
7. A first look at solver performance, which shows that iteration count increases slower than linearly with Courant number, implying that computational speed can be gained from larger time steps (the exception to this is on the hexagonal mesh).

This paper goes further than recent papers presenting advection schemes that are adaptively implicit in the vertical only (Wicker and Skamarock, 2020; Li and Zhang, 2022), not only in the use of implicit time stepping in two dimensions but also improved accuracy while using implicit time stepping and a proof that the mix of implicit and explicit does not destroy boundedness.

Section 1 described multi-tracer efficiency as an essential property of an advection scheme. Using implicit time stepping, a separate solver for each tracer would be necessary, which initially sounds prohibitive. However, much of the cost of an implicit solve is in preconditioning, which would be shared over all tracers that use the same wind field.

Next steps entail three-dimensional solutions and incorporation into a full dynamical core with implicit time stepping for advection in all equations. This will include formulating the MPDATA implicit advection for fully compressible equations considering non-divergent winds and variable density, as well as dynamics right-hand sides, extending the work of Smolarkiewicz and co-workers (Smolarkiewicz and Margolin, 1998; Kühnlein and Smolarkiewicz, 2017).

AUTHOR CONTRIBUTIONS

Hilary Weller: Conceptualization; formal analysis; investigation; methodology; software; validation; visualization; writing – original draft; writing – review and editing. **James Woodfield**: Conceptualization,; formal analysis; investigation; methodology; writing - review and editing. **Christian Kühnlein**: Conceptualization; methodology; writing - review and editing. **Piotr Smolarkiewitz**: Conceptualization; methodology; writing - review and editing.

ORCID

Hilary Weller https://orcid.org/0000-0003-4553-7082

REFERENCES

Baldauf, M. (2019) Local time stepping for a mass-consistent and time-split advection scheme. *Quarterly Journal of the Royal Meteorological Society*, 145, 337–346.

Baldauf, M., Seifert, A., Förstner, J., Majewski, D., Raschendorfer, M. and Reinhardt, T. (2011) Operational convective-scale numerical weather prediction with the COSMO model: description and sensitivities. *Monthly Weather Review*, 139, 3887–3905.

Chen, Y., Weller, H., Pring, S. and Shaw, J. (2017) Comparison of dimensionally split and multi-dimensional atmospheric transport schemes for long time steps. *Quarterly Journal of the Royal Meteorological Society*, 143, 2764–2779.

Cullen, M. and Davies, T. (1991) A conservative split-explicit integration scheme with fourth-order horizontal advection. *Quarterly Journal of the Royal Meteorological Society*, 117, 993–1002.

Davies, T., Cullen, M., Malcolm, A., Mawson, M., Staniforth, A., White, A. and Wood, N. (2005) A new dynamical core for the Met Office's global and regional modelling of the atmosphere. *Quarterly Journal of the Royal Meteorological Society*, 131, 1759–1782.

Dawson, C., Trahan, C.J., Kubatko, E.J. and Westerink, J.J. (2013) A parallel local timestepping Runge-Kutta discontinuous Galerkin method with applications to coastal ocean modeling. *Computer Methods in Applied Mechanics and Engineering*, 259, 154–165.

Dumbser, M., Käser, M. and Toro, E.F. (2007) An arbitrary high-order discontinuous Galerkin method for elastic waves on unstructured meshes - V. Local time stepping and p-adaptivity. *Geophysical Journal International*, 171, 695–717.

Gottlieb, S., Shu, C.-W. and Tadmor, E. (2001) Strong stability-preserving high-order time discretization methods. *SIAM Review*, 43, 89–112.

Harris, L.M., Lauritzen, P.H. and Mittal, R. (2011) A flux-form version of the conservative semi-lagrangian multi-tracer transport scheme (CSLAM) on the cubed sphere grid. *Journal of Computational Physics*, 230, 1215–1237.

Hirt, C., Amsden, A. and Cook, J. (1997) An arbitrary Lagrangian-Eulerian computing method for all flow speeds. *Journal of Computational Physics*, 135, 203–216.

Jebens, S., Knoth, O. and Weiner, R. (2011) Partially implicit peer methods for the compressible Euler equations. *Journal of Computational Physics*.

Kühnlein, C., Deconinck, W., Klein, R., Malardel, S., Piotrowski, Z.P., Smolarkiewicz, P.K., Szmelter, J. and Wedi, N.P. (2019) FVM 1.0: a nonhydrostatic finite-volume dynamical core for the IFS. *Geoscientific Model Development*, 12, 651–676.

Kühnlein, C., Smolarkiewicz, P. and Dörnbrack, A. (2012) Modelling atmospheric flows with adaptive moving meshes. *Journal of Computational Physics*, 231, 2741–2763.

Kühnlein, C. and Smolarkiewicz, P.K. (2017) An unstructured-mesh finite-volume MPDATA for compressible atmospheric dynamics. *Journal of Computational Physics*, 334, 16–30.

Lauritzen, P., Skamarock, W., Prather, M. and Taylor, M. (2012) A standard test case suite for two-dimensional linear transport on the sphere. *Geoscientific Model Development*, 5, 887–901.

Lauritzen, P., Ullrich, P., Jablonowski, C., Bosler, P., Calhoun, D., Conley, A., Enomoto, T., Dong, L., Dubey, S., Guba, O., Hansen, A., Kaas, E., Kent, J., Lamarque, J.-F., Prather, M., Reinert, D., Shashkin, V., Skamarock, W., Sørensen, B., Taylor, M. and Tolstykh, M. (2014) A standard test case suite for two-dimensional linear transport on the sphere: results from a collection of state-of-the-art schemes. *Geoscientific Model Development*, 7, 105–145.

Leonard, B., Lock, A. and MacVean, M. (1996) Conservative explicit unrestricted-time-step multidimensional constancy-preserving advection schemes. *Monthly Weather Review*, 124, 2585–2606.

Li, J. and Zhang, Y. (2022) Enhancing the stability of a global model by using an adaptively implicit vertical moist transport scheme. *Meteorology and Atmospheric Physics*, 134.

Lin, S.-J. (2004) A "vertically Lagrangian" finite-volume dynamical core for global models. *Monthly Weather Review*, 132, 2293–2307.

May, S. and Berger, M. (2017) An explicit implicit scheme for cut cells in embedded boundary meshes. *Journal of Computational Physics*.

Miura, H. (2007) An upwind-biased conservative advection scheme for spherical hexagonal-pentagonal grids. *Monthly Weather Review*, 135, 4038–4044.

Rančić, M., Purser, R. and Mesinger, F. (1996) A global shallow-water model using an expanded spherical cube: gnomonic versus conformal coordinates. *Quarterly Journal of the Royal Meteorological Society*, 122, 959–982.

Smolarkiewicz, P. (1983) A simple positive definite advection scheme with small implicit diffusion. *Monthly Weather Review*, 111, 479–486.

Smolarkiewicz, P. (1984) A fully multidimensional positive definite advection transport algorithm with small implicit diffusion. *Journal of Computational Physics*, 54, 325–362.

Smolarkiewicz, P. and Clark, T. (1986) The multidimensional positive definite advection transport algorithm: further development and applications. *Journal of Computational Physics*, 67, 396–438.

Smolarkiewicz, P. and Grabowski, W. (1990) The multidimensional positive definite advection transport algorithm: nonoscillatory option. *Journal of Computational Physics*, 86, 355–375.

Smolarkiewicz, P. and Margolin, L. (1998) MPDATA: A finite-difference solver for geophysical flows. *Journal of Computational Physics*, 140, 459–480.

Smolarkiewicz, P. and Szmelter, J. (2005) MPDATA: An edge-based unstructured-grid formulation. *Journal of Computational Physics*, 206, 624–649.

Smolarkiewicz, P.K. (2006) Multidimensional positive definite advection transport algorithm: an overview. *International Journal for Numerical Methods in Fluids*, 50, 1123–1144.

Tumolo, G. and Bonaventura, L. (2015) A semi-implicit, semi-Lagrangian discontinuous Galerkin framework for adaptive numerical weather prediction. *Quarterly Journal of the Royal Meteorological Society*, 141, 2582–2601.

Ullrich, P.A., Jablonowski, C., Kent, J., Lauritzen, P.H., Nair, R., Reed, K.A., Zarzycki, C.M., Hall, D.M., Dazlich, D., Heikes, R., Konor, C., Randall, D., Dubos, T., Meurdesoif, Y., Chen, X., Harris, L., Kühnlein, C., Lee, V., Qaddouri, A., Girard, C., Giorgetta, M., Reinert, D., Klemp, J., Park, S.-H., Skamarock, W., Miura, H., Ohno, T., Yoshida, R., Walko, R., Reinecke, A. and Viner, K. (2017) DCMIP2016: a review of non-hydrostatic dynamical core design and intercomparison of participating models. *Geoscientific Model Development*, 10, 4477–4509.

Wicker, L. and Skamarock, W. (2020) An implicit-explicit vertical transport scheme for convection-allowing models. *Monthly Weather Review*, 148, 3893–3910.

Yee, H. (1987) Construction of explicit and implicit symmetric TVD schemes and their applications. *Journal of Computational Physics*, 68, 151–179.

Yee, H., Warming, R. and Harten, A. (1985) Implicit total variation diminishing (TVD) schemes for steady-state calculations. *Journal of Computational Physics*, 57, 327–360.

Zalesak, S. (1979) Fully multidimensional flux-corrected transport algorithms for fluids. *Journal of Computational Physics*, 31, 335–362.

Zerroukat, M. and Allen, T. (2020) SLIC: a semi-lagrangian implicitly corrected method for solving the compressible Euler equations. *Journal of Computational Physics*, 421, 109739.

How to cite this article: Weller, H., Woodfield, J., Kühnlein, C. & Smolarkiewicz, P.K. (2023) Adaptively implicit MPDATA advection for arbitrary Courant numbers and meshes. *Quarterly Journal of the Royal Meteorological Society*, 149(751), 369–388. Available from: https://doi.org/10.1002/qj.4411

APPENDIX A. BOUNDEDNESS OF THE FIRST-ORDER UPWIND ADAPTIVELY IMPLICIT SCHEME

In Section 1.1 we defined a monotonic scheme to be one that "does not generate new spurious extrema or amplify existing extrema". This is usually tested for explicit schemes by checking that solutions at time step $n+1$ are not outside the bounds of solutions at time step n at the same or neighbouring grid boxes. However, this can happen for implicit schemes even if the solution is either preserved exactly or moved and diffused; it should be possible for implicit schemes to move real extrema by more than one grid box in one time step, thus violating the usual test for explicit schemes. Therefore, in order to demonstrate monotonicity of an implicit scheme, we demonstrate global boundedness but without a priori knowledge of the global bounds – whatever the initial global bounds, the solution will remain within them without any tests of what the global bounds are.

The first-order upwind, adaptively implicit scheme can be written thus:

$$\begin{aligned}\psi_c^{n+1} = \psi_c^n + \frac{\Delta t}{\mathcal{V}_c}\sum_{i\in\text{in}}(1-\theta_i)U_i\psi_i^n + \frac{\Delta t}{\mathcal{V}_c}\sum_{i\in\text{in}}\theta_iU_i\psi_i^{n+1} \\ - \frac{\Delta t}{\mathcal{V}_c}\sum_{o\in\text{out}}(1-\theta_o)\,U_o\psi_c^n - \frac{\Delta t}{\mathcal{V}_c}\sum_{o\in\text{out}}\theta_oU_o\psi_c^{n+1},\end{aligned} \tag{A.1}$$

for cell C with faces i and o. Faces "$i \in$ in" have flow into cell C, whereas faces "$o \in$ out" have flow out. Off-centring values are denoted θ_i and θ_o at the different face types. $U_i \geq 0$ and $U_o > 0$ are the inward and outward fluxes. ψ_i are the values of ψ in cells through the i faces. The $\theta_{i,o}$ are defined on faces for conservation. This makes the boundedness of the scheme less straightforward. Equation (A.1) can be rearranged to give

$$\psi_c^{n+1} = \gamma\psi_c^n + \sum_{i\in\text{in}}\alpha_i\psi_i^n + \sum_{i\in\text{in}}\beta_i\psi_i^{n+1}, \tag{A.2}$$

where

$$\alpha_i = \frac{\frac{\Delta t}{\mathcal{V}_c}(1-\theta_i)U_i}{1+\frac{\Delta t}{\mathcal{V}_c}\sum_{o\in\text{out}}\theta_oU_o} \quad \text{for each } i \tag{A.3}$$

$$\beta_i = \frac{\frac{\Delta t}{V_j}\theta_iU_i}{1+\frac{\Delta t}{\mathcal{V}_c}\sum_{o\in\text{out}}\theta_oU_o} \quad \text{for each } i \tag{A.4}$$

$$\gamma = \frac{1-\frac{\Delta t}{\mathcal{V}_c}\sum_{o\in\text{out}}(1-\theta_o)U_o}{1+\frac{\Delta t}{\mathcal{V}_c}\sum_{o\in\text{out}}\theta_oU_o}. \tag{A.5}$$

The quantities α_i, β_i, and γ are all positive as long as the θ_0 are chosen to give

$$\frac{\Delta t}{\mathcal{V}_c}\sum_{o\in\text{out}}(1-\theta_o)U_o \leq 1, \tag{A.6}$$

which can be accomplished by setting

$$\theta_f \geq 1 - \frac{1}{\frac{\Delta t}{\mathcal{V}_c}\sum_{o\in\text{out}}U_o} \tag{A.7}$$

for the cells either side of face f. Hence, all ψ are positive at the next time step. If, in addition, the flow is discretely non-divergent, then

$$\sum_{i\in\text{in}}U_i = \sum_{o\in\text{out}}U_o, \tag{A.8}$$

which implies

$$\sum_{i\in\text{in}}\alpha_i + \sum_{i\in\text{in}}\beta_i + \gamma = 1, \tag{A.9}$$

So, from Equation (A.2), ψ_c^{n+1} is a convex combination of ψ_c^n, ψ_i^n, and ψ_i^{n+1}. This in fact proves that the scheme is globally bounded. This can be shown by contradiction; if we assume that ψ_c^{n+1} is the global maximum at t^{n+1} and it is greater than ψ_j^n for all cells j in the mesh, then Equations A.2 and A.9 cannot both hold for cell C. It is necessary for this scheme to be bounded as it is used as the bounded scheme for the FCT (Section 2.5).

APPENDIX B. STABILITY ANALYSIS OF THE SECOND-ORDER ADAPTIVELY IMPLICIT SCHEME

MPDATA is a nonlinear scheme but the infinite-gauge version is linear, and so von Neumman stability analysis can be applied. In one dimension, for constant velocity $u > 0$, constant Δx, constant θ, and constant Courant number $c = u\Delta t/\Delta x$, the one-dimensional adaptively implicit scheme is

$$\begin{aligned}\psi_j^{n+1} = \psi_j^n - c(1-\theta)(\psi_j^n - \psi_{j-1}^n) - c\theta(\psi_j^{n+1} - \psi_{j-1}^{n+1})\\ - \frac{c}{2}(1-\chi c)(\psi_{j+1}^n - 2\psi_j^n + \psi_{j-1}^n),\end{aligned} \tag{B.1}$$

where ψ_j is ψ at position $x = j\Delta x$. We showed in Section 2.2 that $\chi = 1 - 2\theta$ gives second-order accuracy. From Appendix A we can see that we need $\theta \geq 1 - 1/c$ for stability of the first-order upwind part – the first two terms of Equation (B.1). In order to revert to the explicit version for $c \leq 1$ and to transition smoothly to the implicit version we use

$$\theta = \begin{cases} 0 & c \leq 1 \\ 1 - \frac{1}{c} & c > 1, \end{cases} \tag{B.2}$$

and analyse Equation (B.1) separately for these two cases. Considering a Fourier mode with wave number k, the amplification factor A of Equation (B.1) is

$$A = \frac{1 - c(\chi c - \theta)(1 - \cos k\Delta x) - \mathrm{i}c(1-\theta)\sin k\Delta x}{1 + c\theta(1 - \cos k\Delta x) + \mathrm{i}c\theta \sin k\Delta x}. \tag{B.3}$$

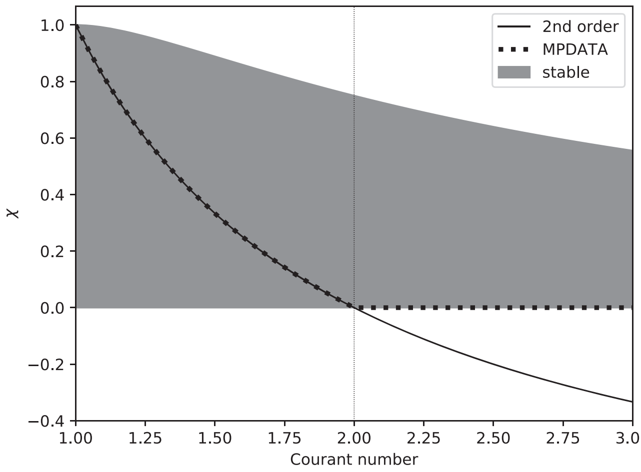


FIGURE B1 Comparison of the stability limits, the second-order requirement and the value of χ used for the MPDATA correction

For $\theta = 0$ and $\chi = 1$ we recover the usual Lax–Wendroff stability constraints of $c \in [-1, 1]$. For $c \geq 1$ and $\theta = 1 - (1/c)$ it can be shown that stability requires $\chi \in [0, (2c-1)/c^2]$. The stability range for χ is compared with the second-order requirement for χ in Figure B1. For behaviour as close as possible to second order for the maximum range of Courant numbers and for stability we use

$$\chi = \max(1 - 2\theta, 0). \tag{B.4}$$

This gives an unconditionally stable scheme with second-order accuracy for $c \leq 2$.