Conservation with moving meshes over orography

Journal of Computational Physics 461 (2022) 111217

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

Conservation with moving meshes over orography

Hiroe Yamazaki [a], Hilary Weller [b,*], Colin J. Cotter [a], Philip A. Browne [c]

[a] Department of Mathematics, Imperial College London, London, UK
[b] Department of Meteorology, University of Reading, Reading, UK
[c] European Centre for Medium-Range Weather Forecasts, Reading, UK

ARTICLE INFO

Article history:
Received 4 August 2021
Received in revised form 7 April 2022
Accepted 7 April 2022
Available online 12 April 2022

Keywords:
Moving meshes
Mesh re-distribution
r-adaptivity
Numerical weather prediction
Orography

ABSTRACT

Adaptive meshes have the potential to improve the accuracy and efficiency of atmospheric modelling by increasing resolution where it is most needed. Mesh re-distribution, or r-adaptivity, adapts by moving the mesh without changing the connectivity. This avoids some of the challenges with h-adaptivity (adding and removing points): the solution does not need to be mapped between meshes, which can be expensive and introduces errors, and there are no load balancing problems on parallel computers. A long standing problem with both forms of adaptivity has been changes in volume of the domain as resolution changes at an uneven boundary. We propose a solution which achieves exact local conservation and maintains a uniform scalar field while the mesh changes volume as it moves over orography. This is achieved by introducing a volume correction parameter which tracks the cell volumes without using expensive conservative mapping.

A finite volume solution of the advection equation over orography on moving meshes is described and results are presented demonstrating improved accuracy for cost using moving meshes. Exact local conservation and maintenance of uniform scalar fields is demonstrated and the correct mesh volume is preserved.

We use optimal transport to generate meshes which are guaranteed not to tangle and are equidistributed with respect to a monitor function. This leads to a Monge-Ampère equation which is solved with a Newton solver. The superiority of the Newton solver over other techniques is demonstrated in the appendix. However the Newton solver is only efficient if it is applied to the left hand side of the Monge-Ampère equation with fixed point iterations for the right hand side.

1. Introduction

Dynamic mesh adaptivity can be advantageous for the numerical solution of PDEs when numerical errors (or their impacts) are greater in some areas than others. Numerical weather and climate predictions, for example, could be improved by locally varying the spatial resolution through time, tracking atmospheric phenomena such as weather fronts [4].

H-adaptivity involves adding and removing computational points based on local resolution requirements (e.g. [1,16,18]). The connectivity of the mesh and the total number of computational points change. Conversely, *r*-adaptivity, or mesh re-distribution, involves moving mesh vertices without changing the connectivity of the mesh. It results in a deformed mesh keeping the number of computational points and topology the same (e.g. [7,9,10]).

* Corresponding author.
E-mail address: h.weller@reading.ac.uk (H. Weller).

https://doi.org/10.1016/j.jcp.2022.111217

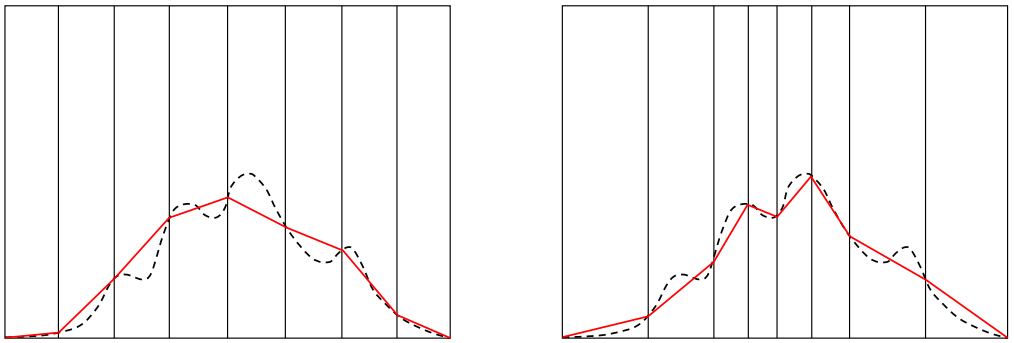

Fig. 1. Two-dimensional example of a change in shape of orography when the mesh distribution changes from (left) uniform to (right) focused in the middle. Solid (red) and dashed lines indicate the model orography and real orography, respectively.

R-adaptivity is an attractive form of adaptivity since the data structures associated with the connectivity do not change and therefore the load balancing remains constant on parallel computers. *R*-adaptivity does not require mapping solutions between old and new meshes. *R*-adaptivity can also lead to smoothly graded meshes, which are desirable in order to reduce wave reflections and other errors associated with rapid resolution changes [17,11].

A disadvantage of *r*-adaptivity is, with a fixed number of points and fixed connectivity, it is not possible to achieve exactly the required resolution in each direction simultaneously, which can be achieved with *h*-adaptivity. This causes an extra difficulty when introducing orography. For example, we consider a vertical slice model with a horizontally moving mesh over orography (Fig. 1). When the topographic surface is approximated as piecewise linear (or higher order) splines based on vertex locations, the shape of orography inevitably changes as mesh vertices move over the orography. Therefore the volume of the domain changes as the mesh moves, which results in unphysical compression or expansion of the model fluid.

With *h*-adaptivity, this issue can be avoided by evaluating all the metric terms from orography on the finest grid everywhere in the domain and average the result, in a consistent manner, to any coarser grid [8]. This approach is only suitable for models in which each fine grid cell only ever over-laps with one coarse grid cell. Another way to resolve this issue is using conservative mapping to calculate the cell volumes over the original shape of orography. For example, Schwartz et al. [14] presented an algorithm to perform highly accurate mappings using sub-grid knowledge, which ensures the required accuracy with grid refinement. Though this approach can be used with both *h*- and *r*-adaptivities, it is expensive to perform conservative mapping of the orography every time step.

Instead of mapping the shape of orography within each cell, we propose another solution which is to correct the cell volumes indirectly by solving a transport equation for a cell volume. We introduce a volume correction parameter which tracks the change in cell volumes caused by the change in the shape of orography. With this approach, the exact local conservation and maintenance of uniform fields on a moving mesh over orography is achieved without using expensive conservative mapping.

Section 2 provides the model description, including the finite-volume discretisation on a moving mesh, and the correction of the cell volumes as mesh moves over orography. In section 3, we present the results of a three-dimensional tracer advection test with the use of the volume correction parameter. Here we evaluate the model error using smooth and rough orography and demonstrate the importance of maintaining uniform fields on a moving mesh. Finally, in section 4 we provide a summary and outlook. We prove that the method for calculating corrected cell volumes is bounded in appendix A and we describe the optimally transported mesh generation in appendix B.

2. Model description

2.1. Finite volume discretisation on a moving mesh

The model description assumes three spatial dimensions but the results (section 3) are all in 2.5 spatial dimensions; two horizontal dimensions and a single mesh layer in the vertical with changes of depth to represent the orography. The test case velocity field is defined in two dimensions using a streamfunction and the initial conditions are two dimensional.

We consider the three-dimensional advection equation in flux form:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\boldsymbol{u}\rho) = 0, \tag{1}$$

where $\boldsymbol{u} = (u, v, w)$ is a prescribed velocity field and ρ is the tracer density. To derive a finite-volume discretised equation, first we integrate the equation over a control volume V and then apply Gauss' divergence theorem:

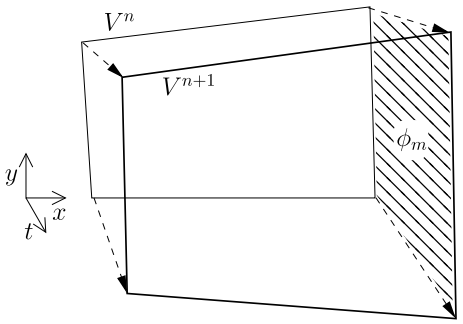

Fig. 2. An example showing how the mesh flux ϕ_m is calculated. The solid line and the thick line show the boundaries of a cell at the time step n and $n+1$, respectively. The hatched region shows the swept volume by the right face of the cell, which is calculated as the mesh flux at time step $n+1/2$ corresponding to that face.

$$\int_V \frac{\partial \rho}{\partial t} \mathrm{d}V + \oint_S \rho \boldsymbol{u} \cdot \boldsymbol{n} \mathrm{d}S = 0, \tag{2}$$

where $\boldsymbol{n}$ is the outward pointing unit normal vector on the boundary surface S of the control volume V so that $\rho \boldsymbol{u} \cdot \boldsymbol{n} \mathrm{d}S$ is the flux of ρ over the surface with area $\mathrm{d}S$. To extend (2) to a moving mesh, we use the Reynolds transport theorem:

$$\frac{\mathrm{d}}{\mathrm{d}t} \int_{V(t)} \rho \mathrm{d}V = \int_{V(t)} \frac{\partial \rho}{\partial t} \mathrm{d}V + \oint_{S(t)} \rho \boldsymbol{u_s} \cdot \boldsymbol{n} \mathrm{d}S, \tag{3}$$

where $\boldsymbol{u_s}$ is the velocity of the boundary surface S. Note that the control volume V and the boundary surface S are now time dependent. The relationship between the volume V and the velocity $\boldsymbol{u_s}$ is called the space conservation law [6]:

$$\frac{\partial}{\partial t} \int_{V(t)} \mathrm{d}V - \oint_{S(t)} \boldsymbol{u_s} \cdot \boldsymbol{n} \mathrm{d}S = 0. \tag{4}$$

This means that the change in the control volume has to match the sum of the swept volumes of all its surfaces. Combining equations (2) and (3), we have the integral form of the advection equation (1) on a moving mesh:

$$\frac{\mathrm{d}}{\mathrm{d}t} \int_{V(t)} \rho \mathrm{d}V + \oint_{S(t)} \rho (\boldsymbol{u} - \boldsymbol{u_s}) \cdot \boldsymbol{n} \mathrm{d}S = 0. \tag{5}$$

The discretised form of the equation (5) can be written:

$$\frac{V^{n+1}\rho^{n+1} - V^n \rho^n}{\Delta t} + \sum_{\text{faces}} \rho_f (\phi - \phi_m) = 0, \tag{6}$$

between time steps n and $n+1$ where values inside the summation are at time step $n+1/2$ (implying second-order accuracy in time), Δt is the time step size, ρ_f denotes the tracer density that is interpolated onto the cell faces, and $\phi = \boldsymbol{u} \cdot \boldsymbol{n} \mathrm{d}S$ is the face flux defined on the cell faces. The spatial accuracy will depend on how ρ_f is evaluated on faces from cell volume data which is described in section 2.4. The mesh flux $\phi_m = \boldsymbol{u_s} \cdot \boldsymbol{n} \mathrm{d}S$ is calculated as the swept volume by the faces during each time step (Fig. 2), which satisfies the following discretised form of the equation (4):

$$\frac{V^{n+1} - V^n}{\Delta t} - \sum_{\text{faces}} \phi_m = 0. \tag{7}$$

2.2. Automatic mesh motion

The optimally transported mesh generation procedure is described in appendix B. The Monge-Ampère equation is solved to generate a mesh that is equidistributed with respect to a monitor function and guaranteed tangle free due to the optimal transport. A Newton method is described to solve the Monge-Ampère equation. A monitor function is chosen so that the cell areas are a factor of 4 smaller in regions where the second derivatives of the tracer density is highest compared with the regions of lowest second derivatives. The factor of 4 is achieved to solver tolerance by the elliptic optimal transport mesh generation, although there are no guarantees about the cell aspect ratios. The mesh is moved every time step so fast convergence of the Newton solver is important, which is also demonstrated in appendix B.

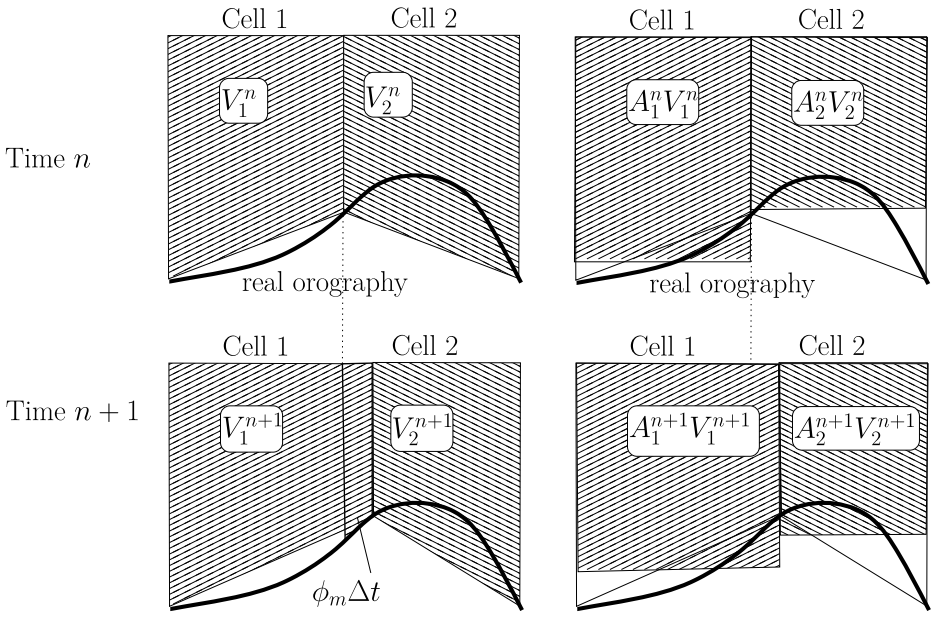

Fig. 3. Two-dimensional example of mesh movement over orography between times n (top) and $n+1$ (bottom). On the left the sum of the volumes changes between n and $n+1$ because of the sampling of the orography. On the right the volumes are multiplying the volume adjustment parameter A, making their sum invariant over time. The thick line represents the surface of orography. Hatched regions show the cell volumes.

The monitor function depends on the second derivative of the tracer density. Higher derivatives are not used because they are more noisy. Even using the second derivative, considerable smoothing is applied to the monitor function (see appendix B.3) as the second-order advection scheme (section 2.4) relies on superconvergence and hence a smoothly varying mesh to achieve second-order accuracy.

2.3. Time stepping with volume adjustment as the mesh moves over orography

When the cell volumes are calculated from vertex locations without tracking the variations in orography within each cell, the shape of the model orography inevitably changes as the mesh vertices move (Fig. 1). This means that equation (7) does not hold on a moving mesh over orography: we only consider the swept volumes in the horizontal, not the swept volume of the orography surface. The mismatch between the change in the control volume and the sum of the swept volumes results in artificial compression and expansion of the fluid.

One solution to this problem would be to use a conservative mapping of old to new mesh to calculate the cell volumes over the original shape of the orography. However we want to avoid the expense of conservative mapping every time step, particularly as we are only interested in the cell volumes and not in the shape of orography itself. Therefore, instead of mapping the shape of orography within each cell, we track the corrected cell volumes by solving the following advection equation for a cell volume:

$$\frac{A^{n+1}V^{n+1} - A^n V^n}{\Delta t} = \sum_{\text{faces}} \tilde{A_f^n} \phi_m, \tag{8}$$

where A is the volume adjustment parameter and V is the volume of cells as defined only by their vertices so that AV corresponds to the corrected cell volumes (Fig. 3). To avoid having negative cell volumes, we use a *downwind* value of A_f^n (with respect to the mesh velocity), denoted by $\tilde{A_f^n}$, in the right-hand side. This guarantees that A is always positive as long as the initial value of A is positive at all cells (see appendix A for a proof). Then we use AV in the advection equation (6) instead of V and solve (6) for ρ^{n+1} using a two stage, second-order Runge-Kutta method:

$$\frac{A^{n+1}V^{n+1}\rho^* - A^n V^n \rho^n}{\Delta t} + \frac{1}{2}\sum_{\text{faces}} \rho_f^n (\phi^n - \tilde{A_f^n}\phi_m) + \frac{1}{2}\sum_{\text{faces}} \rho_f^n (\phi^{n+1} - \tilde{A_f^n}\phi_m) = 0, \tag{9}$$

$$\frac{A^{n+1}V^{n+1}\rho^{n+1} - A^n V^n \rho^n}{\Delta t} + \frac{1}{2}\sum_{\text{faces}} \rho_f^n (\phi^n - \tilde{A_f^n}\phi_m) + \frac{1}{2}\sum_{\text{faces}} \rho_f^* (\phi^{n+1} - \tilde{A_f^n}\phi_m) = 0, \tag{10}$$

where the mesh flux ϕ_m is evaluated at time step $n+1/2$ as in Fig. 2. In this way, we achieve conservation of both the total domain volume (i.e., the total AV) and the total mass relative to the domain size (i.e., the total ρAV), thereby maintaining uniform fields without the need to map the shape of orography within each cell.

The downwinding of A has the effect of smoothing the orography. We will see in section 3.1 that this smoothing does not accumulate, which would make the orography progressively smoother and smoother, but instead it oscillates as the mesh moves to and fro over the orography.

To prove that our scheme preserves a uniform field on a moving mesh, we assume a divergence-free velocity field so that for each cell

$$\sum_{\text{faces}} \phi = 0 \tag{11}$$

at all time steps, and check if the solution stays uniform when the initial condition is uniform. Given $\rho^n \equiv 1$, the equation (9) becomes

$$\frac{A^{n+1}V^{n+1}\rho^* - A^nV^n}{\Delta t} + \frac{1}{2}\sum_{\text{faces}}(\phi^n - \tilde{A_f^n}\phi_m) + \frac{1}{2}\sum_{\text{faces}}(\phi^{n+1} - \tilde{A_f^n}\phi_m) = 0. \tag{12}$$

Substituting the equations (8) and (11) into (12), we have

$$A^{n+1}V^{n+1}(\rho^* - 1) = 0. \tag{13}$$

As $A^{n+1}V^{n+1} \neq 0$, we obtain $\rho^* \equiv 1$. Then equation (10) becomes

$$\frac{A^{n+1}V^{n+1}\rho^{n+1} - A^nV^n}{\Delta t} + \frac{1}{2}\sum_{\text{faces}}(\phi^n - \tilde{A_f^n}\phi_m) + \frac{1}{2}\sum_{\text{faces}}(\phi^{n+1} - \tilde{A_f^n}\phi_m) = 0. \tag{14}$$

In the same way as above, we obtain $\rho^{n+1} \equiv 1$. Therefore it is proved that the solution stays uniform in a divergence-free velocity field when the initial condition is uniform. In section 3, we will confirm this result numerically, whereas the model without the volume adjustment suffers from artificial compression and expansion of the fluid in association with the mesh movement over orography.

2.4. Advection scheme

Section 2.1 describes the interaction of the discretisation with the moving mesh, and section 2.3 includes the description of a two-stage, second-order Runge-Kutta time stepping scheme as in the equations (9) and (10). To complete the discretisation we must specify how face values, ρ_f, are calculated from cell values, ρ. We use a simple, second-order linear upwind advection scheme without monotonicity constraints (e.g. [15]). The use of an unbounded advection scheme makes it easier to ensure that the mesh motion over orography does not generate spurious oscillations. The face values are approximated as:

$$\rho_f = \rho_u + \boldsymbol{\delta} \cdot \nabla_u \rho \tag{15}$$

where ρ_u is the value of ρ in the cell upwind of the face, $\boldsymbol{\delta}$ is the vector that goes from the upwind cell centre to the face centre, and $\nabla_u \rho$ is the gradient of ρ calculated in the upwind cell using Gauss' divergence theorem:

$$\nabla_u \rho = \frac{1}{V}\sum_{\text{faces of } u} \tilde{\rho}_f \mathbf{S}_f \tag{16}$$

where $\tilde{\rho}_f$ is the values of ρ linearly interpolated from cell centres onto faces and $\mathbf{S}_f$ is the outward pointing vector normal to each face with magnitude equal to the face area (the face area vector). This gradient calculation is only second-order accurate on a uniform mesh. We therefore rely on superconvergence for accuracy so the mesh deformations must be smooth which requires smoothing of the monitor function, as described in appendix B.3.

The combination of the two-stage explicit Runge-Kutta time stepping and the linear upwind interpolation means that the advection is stable for Courant numbers up to one.

3. Results

3.1. Advection over smooth and rough orography

In this section, we present the results of an advection test on a three-dimensional mesh with one layer in the vertical. We use a computational domain with a size of $[-L, L] \times [-L, L] \times [0, H]$, where the domain half-length L and height H are

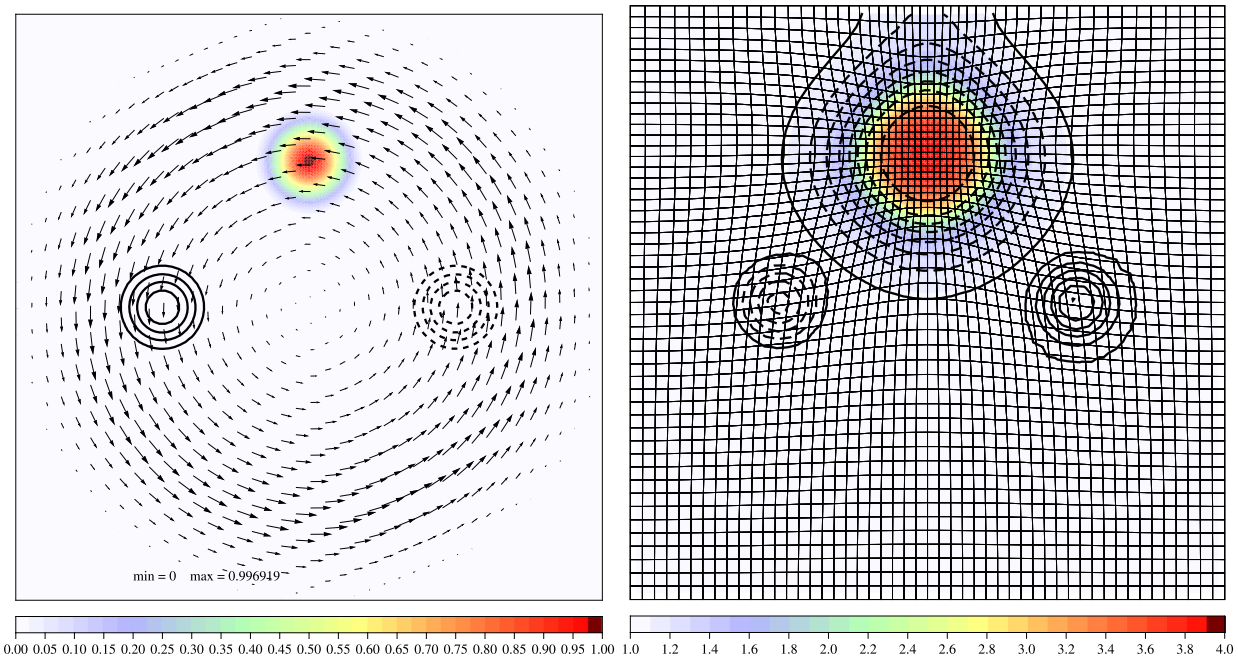

Fig. 4. Left: initial conditions of the tracer density (coloured/shaded), the velocity and height of orography (dashed is negative, contour interval 100 m). Right: the monitor function (coloured/shaded), the resulting 50×50 mesh and the cell volumes (contoured). The Monge-Ampère mesh solution ensures that the product of the monitor function and the cell volumes is uniform.

set to 5 km and 1 km, respectively. The number of cells is N both in the x and y directions. All boundaries of the domain are considered as rigid walls.

An initial tracer (coloured/shaded in Fig. 4a), is transported by a solid body rotating velocity field. The initial tracer density ρ_0 is defined as

$$\rho_0(x, y) = \begin{cases} \frac{1}{2}\left[1 + \cos\left(\frac{\pi r_t}{R_t}\right)\right] & (r_t \le R_t) \quad \text{(a)} \\ 0 & (r_t > R_t) \quad \text{(b)} \end{cases} \tag{17}$$

where the tracer radius $R_t = L/5$, and the distance to the centre of the tracer,

$$r_t = |\mathbf{x} - \mathbf{x}_t| = \sqrt{(x - x_t)^2 + (y - y_t)^2}, \tag{18}$$

with the centre of the tracer initially at $\mathbf{x}_t = (0, L/2)$. The divergence-free velocity field can be written in terms of a streamfunction ψ as

$$u = -\frac{\partial \psi}{\partial y}, \quad v = \frac{\partial \psi}{\partial x}, \quad w = 0. \tag{19}$$

We use ψ that yields a velocity field which rotates around the centre of the domain and decays linearly to zero before it reaches the boundaries (Fig. 4a):

$$\psi(x, y) = \begin{cases} \Omega\, r_v^2 & (r_v \le R_i) \quad \text{(a)} \\ \Omega\, R_i \left\{ R_i + (r_v - R_i)\left(\frac{R_o - r_v}{R_o - R_i} + 1\right)\right\} & (R_i < r_v \le R_o) \quad \text{(b)} \\ \Omega R_i R_o & (r_v > R_o) \quad \text{(c)} \end{cases} \tag{20}$$

where r_v is the distance to the centre of the domain. The inner radius is set to $R_i = 0.76L$ so that the tracer is separate from the sheared velocity and the outer radius is set to $R_o = L$ so that the velocity is zero at the boundary. The angular velocity is given by $\Omega = \pi/600$ s^{-1} so that the tracer is transported counterclockwise and reaches its initial position after 600 seconds. Note that, although the velocity field is steady, it is recalculated after every time step since the mesh is changing.

The streamfunction is evaluated at vertices on the three-dimensional mesh and the fluxes are calculated using Stoke's circulation theorem around faces (between cells). This ensures that the velocity field is discretely divergence free. Note that

the velocity flux, ϕ, is not scaled by A as this would create divergence. A acts on the mesh fluxes and volumes but not on the fluid fluxes.

The tracer passes over a smooth hill and valley, as shown in Fig. 4a, with surface height given by

$$h(x,y) = \begin{cases} \frac{h_{\max}}{2}\left[1+\cos\left(\frac{\pi r_h}{a}\right)\right] & (r_h \le a) \quad \text{(a)} \\ \frac{h_{\min}}{2}\left[1+\cos\left(\frac{\pi r_v}{a}\right)\right] & (r_v \le a) \quad \text{(b)} \\ 0 & (r_h > a \text{ and } r_v > a) \quad \text{(c)} \end{cases} \tag{21}$$

where $a = L/5$ is the orography radius, $h_{\max} = 500$ m is the height at the centre of the hill and that of the valley $h_{\min} = -500$ m. The distance to the centre of the hill $r_h = |\mathbf{x} - \mathbf{x}_h|$ with $\mathbf{x}_h = (-L/2, 0)$, and the distance to the centre of the valley $r_v = |\mathbf{x} - \mathbf{x}_v|$ with $\mathbf{x}_v = (L/2, 0)$.

As a more challenging test, the tracer also passes over a cylinder-shaped hill and valley which has steep cliffs on the sides, with surface height given by

$$h(x,y) = \begin{cases} h_c & (r_h \le a) \quad \text{(a)} \\ -h_c & (r_v \le a) \quad \text{(b)} \\ 0 & (r_h > a \text{ and } r_v > a) \quad \text{(c)} \end{cases} \tag{22}$$

where $h_c = 500$ m.

The monitor function calculated from the initial tracer field is coloured/shaded in Fig. 4b along with an initial 50×50 mesh calculated from this monitor function. The cell volumes are contoured which shows that the Monge-Ampère mesh solution ensures the volumes closely fit the requirements. However cells can be stretched in order to achieve the required volume and there is no control over the cell aspect ratios. The contoured cell volumes in Fig. 4b also show the smooth mountain (on the left) and the smooth valley (on the right). The mesh generation controls cell volumes before the orography is created so that the volume changes due to the orography do not influence the horizontal resolution. i.e. the monitor function controls the horizontal cell areas.

The control run over smooth orography uses the volume correction, 100×100 cells and a time step of $\Delta t = 0.5$ s. The tracer after a quarter, half, three quarters and a full revolution of the control run are coloured/shaded in Fig. 5 and the monitor function (indicating the mesh density) is contoured. As the velocity field is divergence-free, the tracer is accelerated over the hill and decelerated over the valley, returning to its original shape and position after 600 s. The mesh tracks the tracer as it moves and changes its shape, without tangling. As described in section 2.2, the monitor function is chosen here so that the cell areas are a factor of 4 smaller in regions where the second derivatives of the tracer density is highest compared with the regions of lowest second derivatives (see appendix B.3 for details). The results with and without the volume correction, A, look visually identical for the smooth orography (not shown).

Fig. 6a,b shows the tracer advection over the rough orography at a quarter and one whole revolution without using volume correction, A. Large oscillations are generated due to the compression and expansion of the domain. These oscillations are not always visible in Fig. 6a,b because the tracer is only coloured between zero and one but the location of the largest monitor function shows where the oscillations are large. It might be possible to dampen the oscillations with an aggressive monotonic advection scheme but monotonic advection should only be achieved for non-divergent wind fields whereas we have introduced artificial divergence by changing the mesh volume.

The control run over the rough orography using the volume correction, A, gives very similar results to the results over smooth orography (Fig. 6c,d), demonstrating the success of correcting the volumes and the importance of the maintenance of uniform fields on a moving mesh over orography.

3.2. Correcting the volume and preserving uniform fields

Fig. 7 shows how the total, uncorrected domain changes in volume as the mesh moves over the smooth and rough orography over five revolutions of the solid body rotation for the control runs solving the advection equation with A. The rough orography creates larger and more rapidly varying changes in domain volume than the smooth orography. The corrected mesh, with cell volumes AV, does not change size to machine precision, as required. We also require that A does not deviate too far from one regardless of the length of the simulation as this would mean that the orography is getting progressively smoother. This is of particular concern as we know that A is calculated using first-order numerics. The maximum and minimum values and the variance of A for ten revolutions are shown in Figs. 8 and 9 for the smooth and rough orography, confirming that deviations from one do not accumulate.

To demonstrate that the model with A maintains uniform fields as the mesh moves over orography, we repeat the experiment using a uniform initial condition, $\rho_0 \equiv 1$, instead of using the cosine-shaped tracer but the mesh moves to track the original cosine-shaped tracer. We compare results with and without the volume correction, A. Figs. 10a,b,d,e show the initially uniform tracer density at $t = 150$ s and $t = 600$ s without the volume correction. For the smooth orography there is

Fig. 5. Tracer advection over smooth orography with a moving mesh correcting for volume errors. Colours/shading show the tracer density at $t =$ (a) 150 s, (b) 300 s, (c) 450 s and (d) 600 s. Solid and dashed black contours show the positive and negative height of orography, with a contour interval of 100 m. Grey contours show the monitor function which is inversely proportional to the horizontal cell area. Resolution 100×100 cells, $\Delta t = 0.5$ s.

artificial compression and expansion of the fluid leading to deviations of the tracer from one. For the rough orography, the deviations from one are much larger, with shorter wavelengths and fill the whole of the rotating domain. Figs. 10c,f show $10^{12}(\rho - 1)$ at $t = 600$ s when using A which shows that the uniform ρ is maintained to machine precision for the smooth and rough orography. The values of A calculated are shown in Fig. 11.

3.3. Convergence with resolution

Advection over flat, smooth and rough orography is simulated using the model with volume corrections and with a range of resolutions, varying the time step in proportion to the resolution, as shown in Table 1. The adaptive mesh simulations are

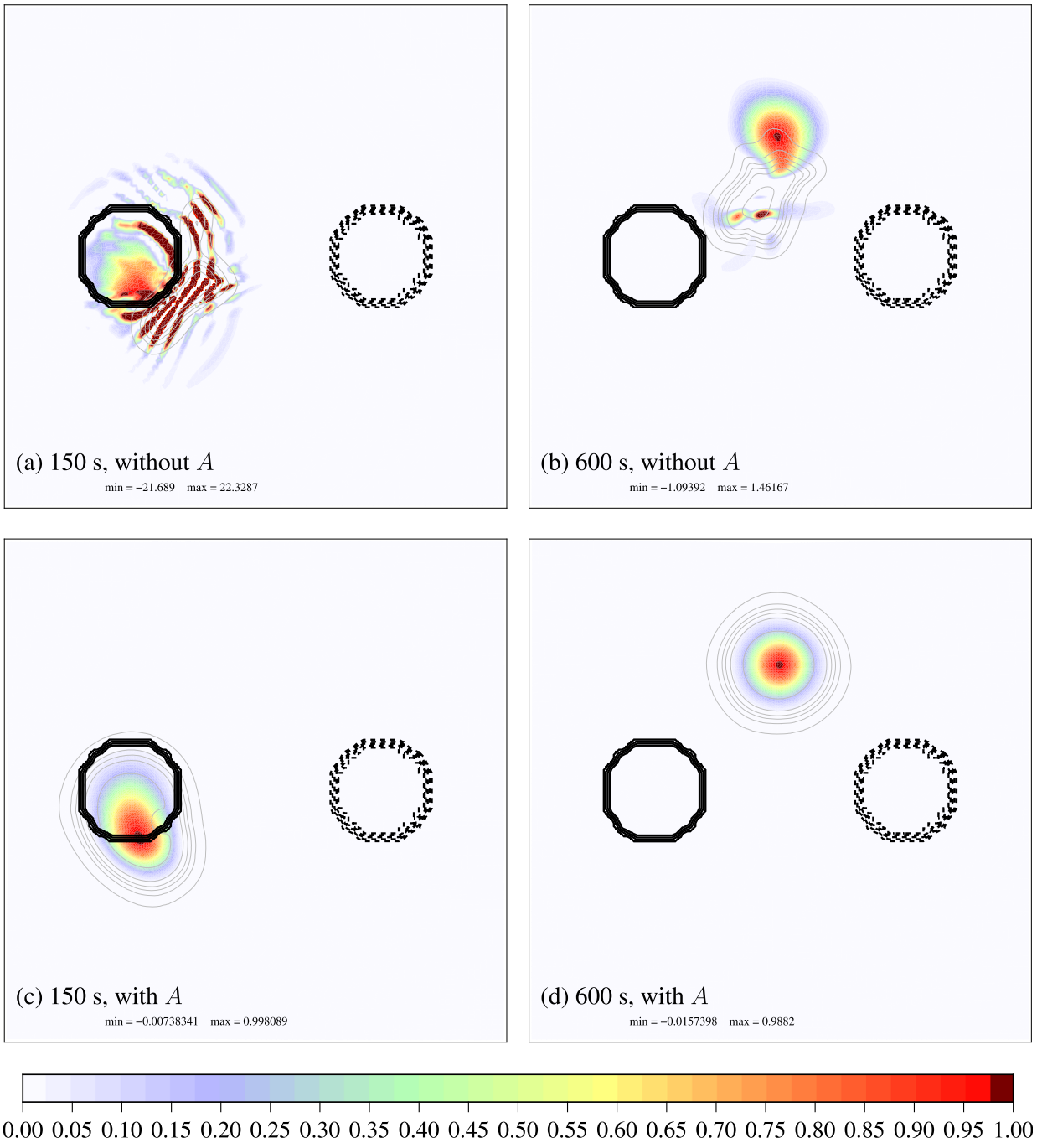

Fig. 6. Tracer advection over rough orography with a moving mesh without (a,b) and with (c,d) correcting for volume errors with A. Colours/shading show the tracer density at $t =$ (a,c) 150 s, (b,d) 600 s. Solid and dashed black contours show the positive and negative height of orography, with a contour interval of 100 m. Grey contours show the monitor function which is inversely proportional to the horizontal cell area. Resolution 100×100 cells, $\Delta t = 0.5$ s.

compared with fixed mesh simulations. The fixed and adaptive simulations use the same time step for the same nominal resolution so the maximum Courant number is larger for the adaptive simulations due to the local refinement.

The ℓ_2 error norms versus nominal (average) resolution after one revolution for fixed and moving meshes are shown in Fig. 12. The moving mesh reduces errors by about a factor of four both with and without orography. The presence or shape of the orography makes little difference to the errors which explains why the smoothing of the orography during the volume correction does not affect the errors. Simulations on the fixed, uniform mesh converge with a rate of 1.78, close to the theoretical convergence rate of two. Although the errors are lower with mesh adaptation, the order of convergence is also slightly lower at a rate of 1.64. This is likely to be because the mesh is no longer uniform and so the solution

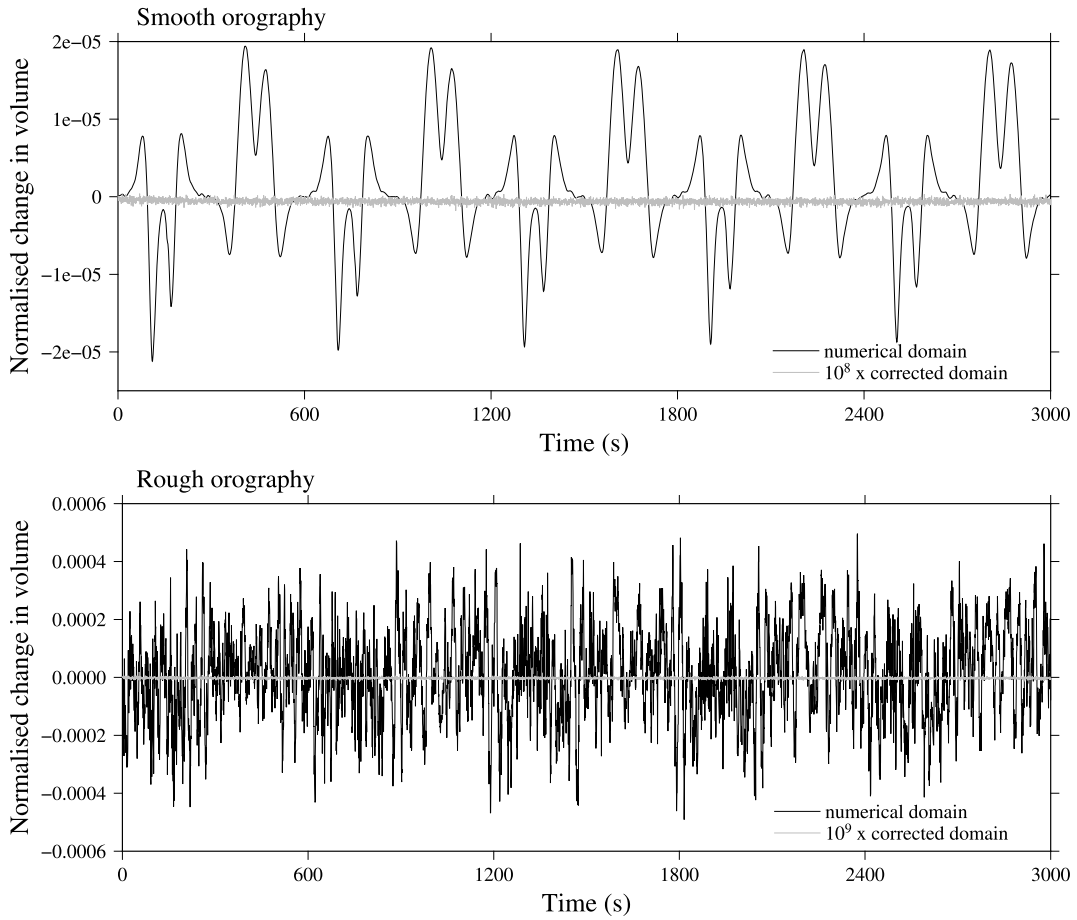

Fig. 7. The normalised change in volume of the domain when moving the 100×100 cell mesh over smooth (top) and rough (bottom) orography for five full revolutions of the solid body rotation and the corrected volume. The corrected domain changes are multiplied by 10^8 for the smooth orography and by 10^9 for the rough orography.

Table 1
Resolution, time step and maximum Courant number statistics for the convergence with resolution tests using fixed and moving meshes.

No. cells	Δx	Δx adapt		Δt	max C	max C
	fixed	min	max		fixed	adapt
50×50	200	104	292	1	0.27	0.52
100×100	100	50	162	0.5	0.28	0.53
200×200	50	25	85	0.25	0.28	0.83
400×400	25	12.5	46	0.125	0.28	0.88

benefits less from superconvergence. The sub-optimal convergence might be improved by optimising the monitor function and would certainly be improved by using a higher-order advection scheme, both of which are outside the scope of this paper.

4. Conclusion

We proposed a novel approach to solve the problem of changes in volume of the domain when resolution changes over orography in a simulation using adaptive meshes. The volume correction parameter is introduced which tracks the cell volumes by solving an advection equation for a cell volume, achieving conservation of both the volume of the domain and the total mass without expensive conservative remapping. The results of tracer advection test showed that our scheme maintains a uniform field while the mesh resolution changes over orography, whereas the model without the volume correction suffers from artificial compression and expansion of the fluid due to the lack of conservation of volume of the

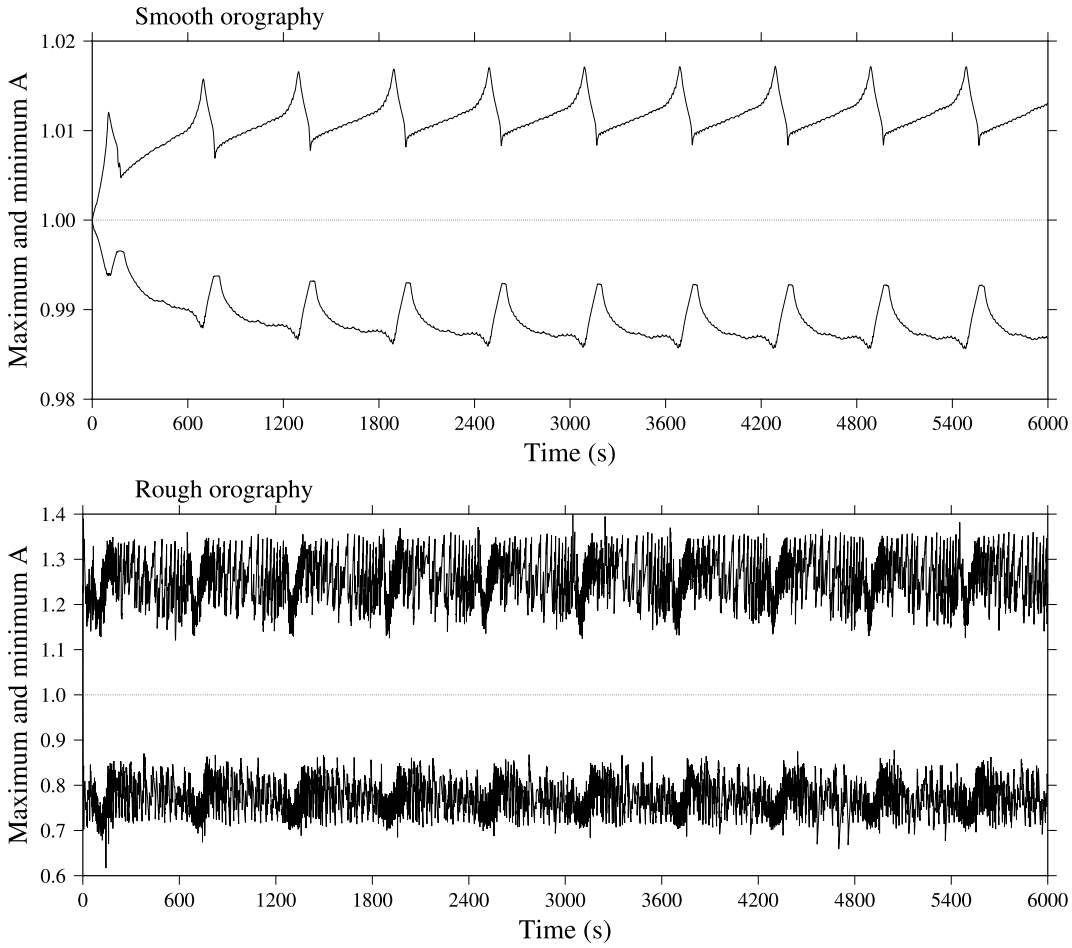

Fig. 8. The maximum and minimum of the volume correction parameter, A, when moving the 100×100 cell mesh over smooth and rough orography for ten full revolutions of the solid body rotation.

domain. The importance of the maintenance of uniform fields was demonstrated over rough orography where the change in cell volumes on a moving mesh can be pronounced without volume correction. The resulting artificial changes in volume lead to large unbounded errors when the advected tracer moves over orography. The volume correction successfully avoided the errors by efficiently tracking the changes in the cell volumes over orography and correcting the cell volumes. The same idea is considered to be applicable to other variable boundary conditions on a moving mesh (e.g. a land sea mask). The volume correction effectively smooths the orography and the smoothing reaches an oscillating steady state which depends on the roughness of the original orography. Further work is needed to apply this method to the shallow water or fully compressible equations with the aim of simulating atmospheric problems on a moving mesh over real orography.

The optimally transported mesh is moved by solving the Monge-Ampère equation at every time step. A novel Newton solve of the Monge-Ampère equation is described in the appendix.

CRediT authorship contribution statement

Hiroe Yamazaki: Conceptualisation, Preparation of original draft, Method, Code, Visualisation, Investigation.
Hilary Weller: Conceptualisation, Preparation of revised draft, Method, Code, Visualisation, Investigation.
Colin J. Cotter: Conceptualisation, Method, Manuscript review.
Philip A. Brown: Conceptualisation, Method, Manuscript review.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

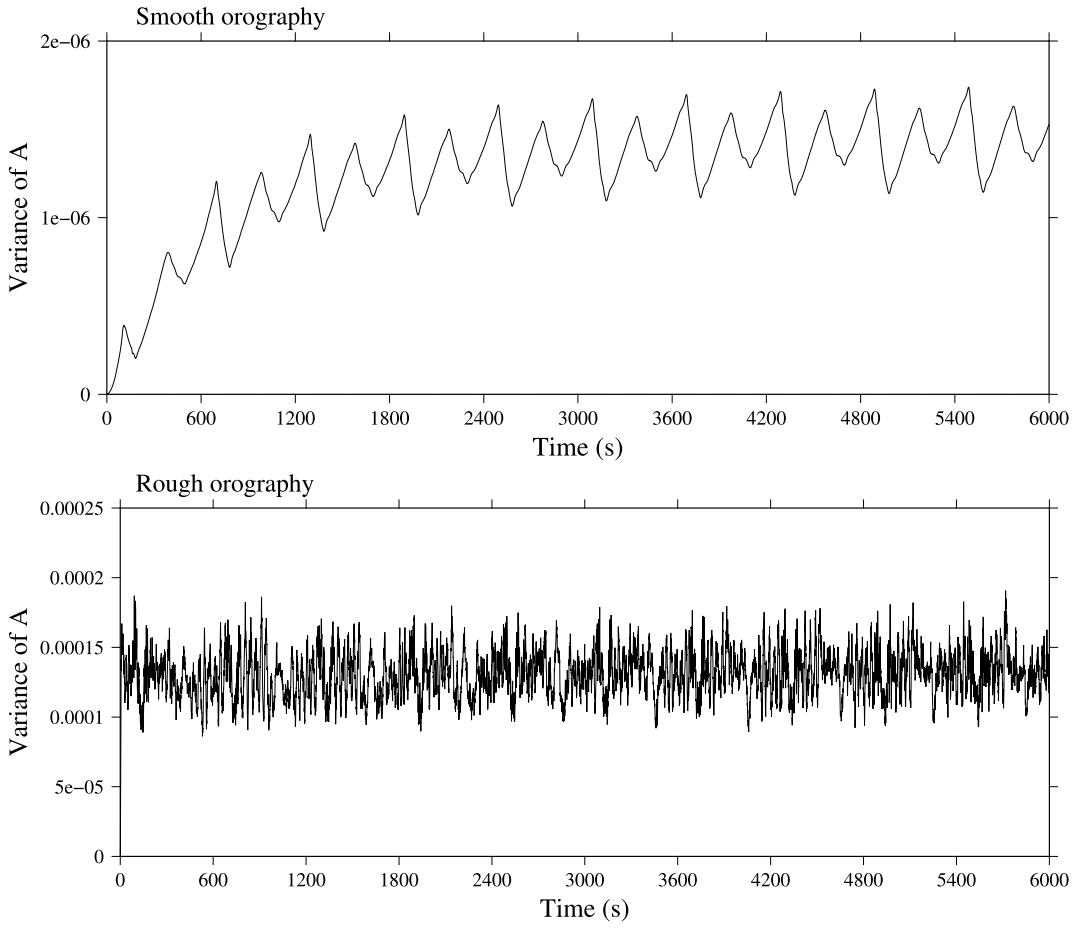

Fig. 9. The variance of the volume correction parameter, A, when moving the 100×100 cell mesh over smooth and rough orography for ten full revolutions of the solid body rotation.

Acknowledgement

We would like to acknowledge NERC grant NE/M013693/1. CJC additionally acknowledges support from NERC grant NE/M013634/1. The source code for this project is located at https://github.com/AtmosFOAM/AMMM with the tag paper.2021; the simulation code itself is located within this repository under run/advection/advectionOTFoam/advectionOT/runAll. The codes are developed with OpenFOAM 7 (https://openfoam.org, 2019).

Appendix A. Boundedness of volume adjustments

In section 2.3, we introduced a volume adjustment parameter, A, to correct the cell volumes, V, calculated from vertex locations sampled over orography. As the mesh moves, the sum of all V changes but the sum of AV does not change. Here we prove that A is bounded above zero which is needed to guarantee that the model may not have negative cell volumes.

The parameter A is calculated from an advection equation (8) discretised using first-order forward in time and first-order downwind in space, which can be rewritten as:

$$\begin{aligned}\frac{A^{n+1}V^{n+1} - A^n V^n}{\Delta t} &= \sum_{\substack{\text{outward}\\\text{faces}}} A_N^n \underbrace{\phi_m}_{\text{positive}} + \sum_{\substack{\text{inward}\\\text{faces}}} A^n \underbrace{\phi_m}_{\text{negative}} \\ &= \sum_{\substack{\text{outward}\\\text{faces}}} A_N^n \underbrace{\phi_m}_{\text{positive}} - \sum_{\substack{\text{inward}\\\text{faces}}} A^n \underbrace{|\phi_m|}_{\text{positive}}, \end{aligned} \tag{23}$$

where A_N^n denotes the tracer density at the neighbouring cell downstream. This can be re-arranged for A^{n+1} as a function of values of A at time level n:

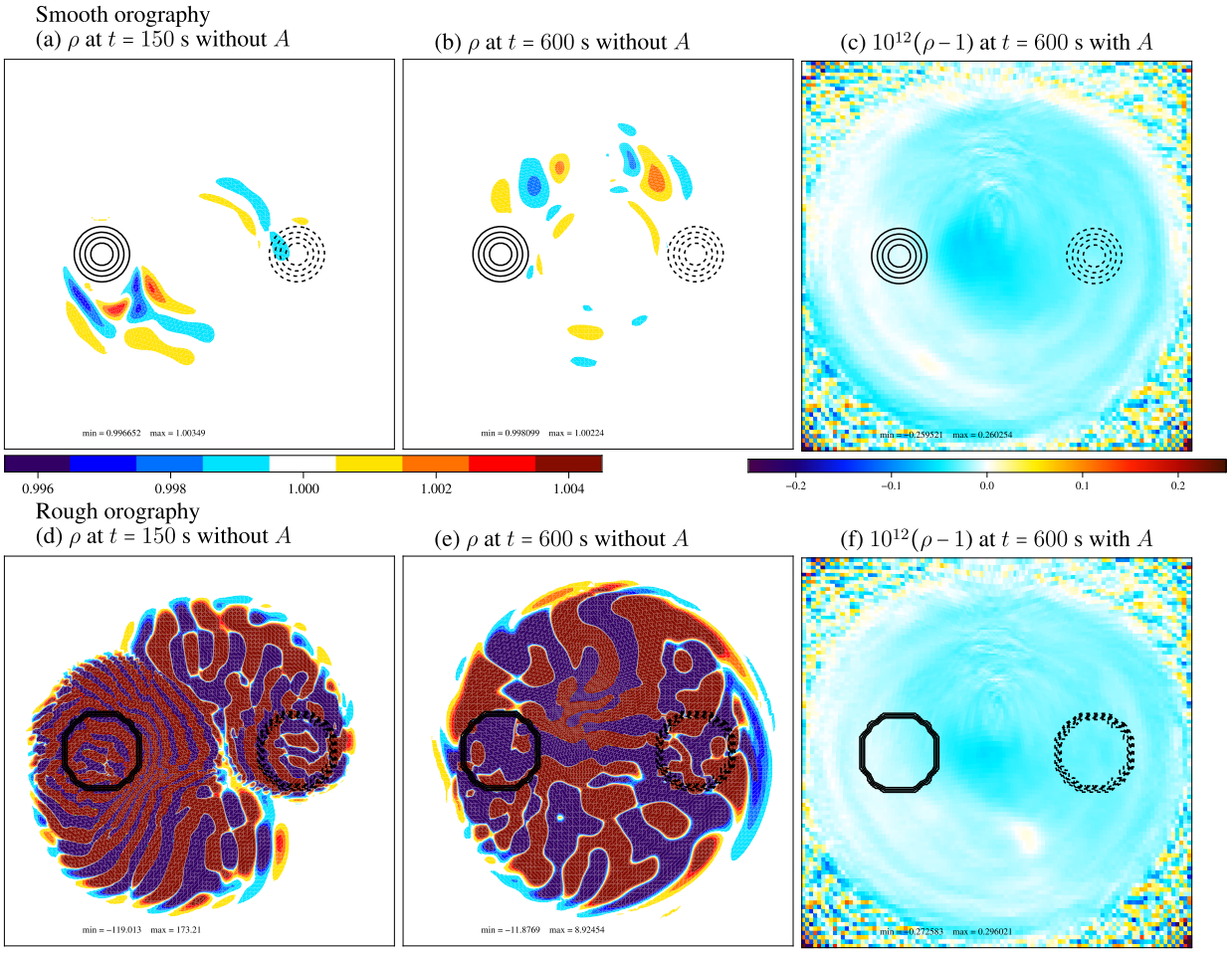

Fig. 10. Advection of $\rho \equiv 1$ over smooth and rough orography with mesh movement to track the cosine hill tracer. a,b,d,e do not use the volume corrector, A. c,f uses A and the difference between ρ and 1 multiplied by 10^{12} is plotted.

$$A^{n+1} = \frac{V^n}{V^{n+1}} A^n \left\{ 1 - \frac{\Delta t}{V^n} \sum_{\substack{\text{inward} \\ \text{faces}}} \underbrace{|\phi_m|}_{\text{positive}} \right\} + \frac{\Delta t}{V^{n+1}} \sum_{\substack{\text{outward} \\ \text{faces}}} A_N^n \underbrace{\phi_m}_{\text{positive}}. \tag{24}$$

Given that $A^n > 0$ and $A_N^n > 0$, we can see that $A^{n+1} > 0$ when

$$1 - \frac{\Delta t}{V^n} \sum_{\substack{\text{inward} \\ \text{faces}}} \underbrace{|\phi_m|}_{\text{positive}} = 1 - \underbrace{\left| \frac{\Delta t}{V^n} \sum_{\substack{\text{inward} \\ \text{faces}}} \phi_m \right|}_{\text{positive}} > 0. \tag{25}$$

Since the mesh velocity Courant number, C, is defined as

$$C = \frac{\Delta t}{V^n} \sum_{\text{faces}} \phi_m, \tag{26}$$

$A^{n+1} > 0$ if $|C| < 1$. Therefore it is proved that, when the initial value of A is positive at all cells, A stays positive as long as the mesh velocity Courant number is less than one.

Appendix B. Numerical solution of the Monge-Ampère equation for mesh generation

The meshing technique is described in full, analysed and compared with other methods full in [3] which is summarised here and results are presented for the meshes used in this paper.

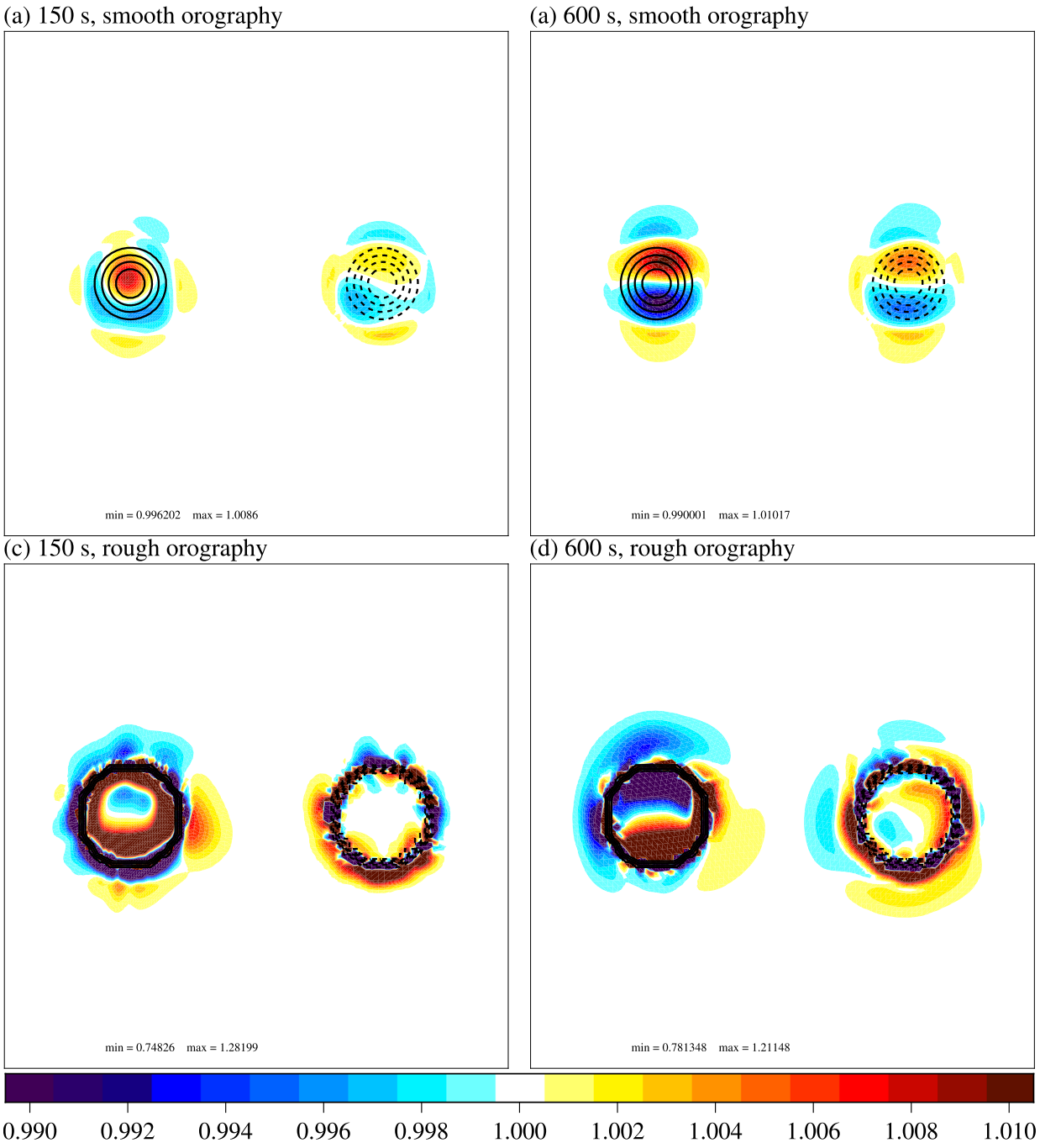

Fig. 11. A at t = (a) 150 s and (b) 600 s in the advection test over smooth and rough orography. Solid and dashed contours show the profile of orography as in Fig. 10.

B.1. Introduction

An optimally transported mesh is as close as possible to the original mesh (close being defined by the root mean square distance between the vertices of the original and transported mesh) whilst equidistributing a given scalar monitor function [5]. To guarantee that the transported mesh is not tangled, the locations, $\boldsymbol{x}$, are defined from the locations of the original mesh, $\boldsymbol{\xi}$, by the addition of the gradient of a mesh potential, ϕ:

$$\boldsymbol{x} = \boldsymbol{\xi} + \nabla\phi. \tag{27}$$

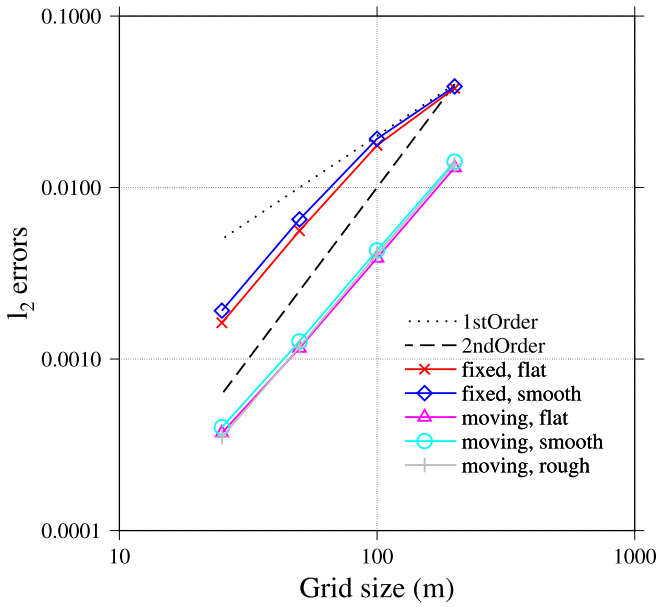

Fig. 12. The ℓ_2 error norm in ρ versus the grid size for the advection test over smooth orography using a cosine-shaped tracer. The fixed, uniform mesh solutions converge at a rate of 1.78 both with and without the smooth orography. The moving mesh simulations over flat, smooth (mountains) and rough (cliffs) orography converged at a rate of 1.64.

Equidistribution of the monitor function, $m(\boldsymbol{x}) > 0$ is expressed as:

$$|\nabla \boldsymbol{x}|\, m(\boldsymbol{x}) = c \tag{28}$$

for a constant c uniform across space where $|\ |$ is the matrix determinant so that $|\nabla \mathbf{x}|$ is the Jacobian determinant of the map from the original to the transported mesh. The combination of equations (27) and (28) gives a fully non-linear elliptic PDE, the Monge-Ampère equation:

$$|I + H(\phi)| = \frac{c}{m(\boldsymbol{x})} \tag{29}$$

where $I = \nabla \boldsymbol{\xi}$ is the identity tensor and $H = \nabla\nabla$ is the Hessian. The meshes in this paper are all the result of numerical solution of the Monge-Ampére equation.

Budd and Williams [5] added Laplacian smoothing and a rate of change term to (29) making it parabolic and solved using a spectral method. Weller et al. [19] derived an equation to generate optimally transported meshes on the surface of a sphere, linearised about a uniform flat mesh to create fixed point iterations, each iteration requiring the solution of a Poisson equation discretised using finite volumes. McRae et al. [12] re-wrote the equation on the surface of a sphere as a PDE and solved using a Newton solver with finite elements. Here we describe a Newton method for solving the Monge-Ampère equation on a finite plane and discretise in space with finite volumes following [19].

B.2. Numerical method

We define a Newton method for solving (29) in Euclidean geometry, linearising the LHS around the previous iteration and using the RHS from the previous iteration. $\boldsymbol{x}^k = \boldsymbol{\xi} + \nabla\phi^k$ is the solution at iteration k. By writing $\phi^{k+1} = \phi^k + \varepsilon\psi$ it can be shown that

$$|I + H(\phi^{k+1})| = |I + H(\phi^k)| + \nabla \cdot P^k \nabla \varepsilon\psi + \mathcal{N}(\varepsilon\psi), \tag{30}$$

where P^k is the matrix of cofactors of $I + H(\phi)$ and $\mathcal{N}$ is some nonlinear function. In 2D

$$P^k = \begin{bmatrix} 1 + \phi^k_{yy} & -\phi^k_{xy} \\ -\phi^k_{xy} & 1 + \phi^k_{xx} \end{bmatrix} \tag{31}$$

and $\mathcal{N}(\varepsilon\psi) = \varepsilon^2 |H(\psi)|$. In 3D, a more involved calculation can show $\mathcal{N}(\varepsilon\psi) = \varepsilon^3 \tilde{\mathcal{N}}(\psi)$ and

$$P^k = \begin{bmatrix} 1+\phi^k_{yy}+\phi^k_{zz}+\phi^k_{yy}\phi^k_{zz}-\phi^k_{yz}\phi^k_{yz} & -\phi^k_{xy}-\phi^k_{xy}\phi^k_{zz}+\phi^k_{xz}\phi^k_{yz} & -\phi^k_{xz}-\phi^k_{xz}\phi^k_{yy}+\phi^k_{xy}\phi^k_{yz} \\ -\phi^k_{xy}-\phi^k_{xy}\phi^k_{zz}+\phi^k_{xz}\phi^k_{yz} & 1+\phi^k_{xx}+\phi^k_{zz}+\phi^k_{xx}\phi^k_{zz}-\phi^k_{xz}\phi^k_{xz} & -\phi^k_{yz}-\phi^k_{xx}\phi^k_{yz}+\phi^k_{xy}\phi^k_{xz} \\ -\phi^k_{xz}-\phi^k_{xz}\phi^k_{yy}+\phi^k_{xy}\phi^k_{yz} & -\phi^k_{yz}-\phi^k_{xx}\phi^k_{yz}+\phi^k_{xy}\phi^k_{xz} & 1+\phi^k_{xx}+\phi^k_{yy}+\phi^k_{xx}\phi^k_{yy}-\phi^k_{xy}\phi^k_{xy} \end{bmatrix}.$$

At convergence terms proportional to ε^d (where d is the dimensionality of space) will disappear so at each iteration, we solve the following Poisson equation for $\varepsilon\psi$:

$$\nabla\cdot\left(P^k\nabla\varepsilon\psi\right) = \frac{c}{m(\boldsymbol{x}^k)} - |I + H(\phi^k)|. \tag{32}$$

Equation (32) is elliptic as long as P^k is positive definite. For simplicity and efficiency we use a finite volume discretisation for spatial discretisation of (32). However, unlike wide stencil finite difference methods (e.g. [13]), this is not guaranteed to give monotonic solutions. Wide stencil finite difference methods increase the stencil size until the determinant of the Hessian in the Monge-Ampère equation is positive. They are quite expensive and we are taking a more pragmatic approach. Our approach means that P^k can become non-positive definite so (32) loses its ellipticity and solutions rapidly diverge. To remedy this we can modify (32) to maintain ellipticity by replacing the matrix P^k with a modified matrix Q^k such that

$$Q^k = P^k + \gamma I \tag{33}$$

and γ is defined as

$$\gamma := \begin{cases} 0 & \text{if} \quad \min\sigma[P^k] > 0 \\ \delta - \min\sigma[P^k] & \text{if} \quad \min\sigma[P^k] \le 0. \end{cases} \tag{34}$$

The constant $\delta > 0$ is chosen to avoid round-off errors (we have taken $\delta = 10^{-5}$), and $\sigma[P^k]$ refers to the spectrum of P^k. This process simply shifts the eigenvalues of the matrix P^k so that they remain positive.

The iterations labelled k are called outer iterations because the Poisson equation is also solved using an iterative solver within each outer iteration.

The Laplacian and the Hessian of (32) are discretised in space using compact finite volumes, following [19]. This is equivalent to second order finite differences on a uniform grid. Zero gradient boundary conditions are used. The spatial discretisation leads to a set of linear simultaneous equations. These are solved using the OpenFOAM GAMG solver with a symmetric Gauss Seidel smoother and an LU pre-conditioner. A maximum of 10 solver iterations are allowed. The tightest solver tolerance is 10^{-4} but the solver is only solved to a tolerance of 0.01 times the initial residual each outer iteration. This is to avoid spending too much time solving the first few iterations tightly when subsequent iterations will have updated coefficients. Tightening the tolerance would reduce the small discrepancies between the mesh volumes and c/m.

B.3. The monitor function

The monitor function is based on the Frobenius norm of the Hessian of the tracer density, ρ, which in two dimensions is

$$m_1(\boldsymbol{x}) = \sqrt{\rho^2_{xx}+\rho^2_{xy}+\rho^2_{yx}+\rho^2_{yy}}. \tag{35}$$

Following McRae et al. [12] we use the rule of thumb that half of the resolution should be placed where not much is happening. This can be approximately achieved by setting:

$$m_2(\boldsymbol{x}) = \min\left(\frac{m_1}{\overline{m_1}+1}, r_{\max}\right) \tag{36}$$

where $\overline{m_1}$ is the area average of m_1 and $r_{\max}$ is the ratio of smallest to largest cell volumes/areas of the resulting adapted mesh. For the simulations in section 3 we use $r_{\max} = 4$ meaning that, if cells have aspect ratio 1 then the maximum ratio of smallest to largest cell side lengths is 2. The monitor function is smoothed before it is used for mesh generation so that the resulting mesh varies smoothly, which is advantageous for finite volume and finite difference methods that have the property of super convergence. The final monitor function, m_3, is the implicit solution of the diffusion equation:

$$\frac{m_3 - m_2}{\Delta t} = K\nabla^2 m_3 \tag{37}$$

where the diffusion coefficient, K, is mesh size and time step dependent:

$$K = M\frac{\Delta x^2}{4\Delta t}. \tag{38}$$

M is equivalent to the number of applications of a $(1,-2,1)$ filter to smooth the monitor function. $M = 20$ is used for the meshes presented in section 3. The Laplacian in (37) is calculated on the uniform orthogonal computational mesh of squares using 2nd-order centred differences (i.e. $(1,-2,1)$ differencing in each direction).

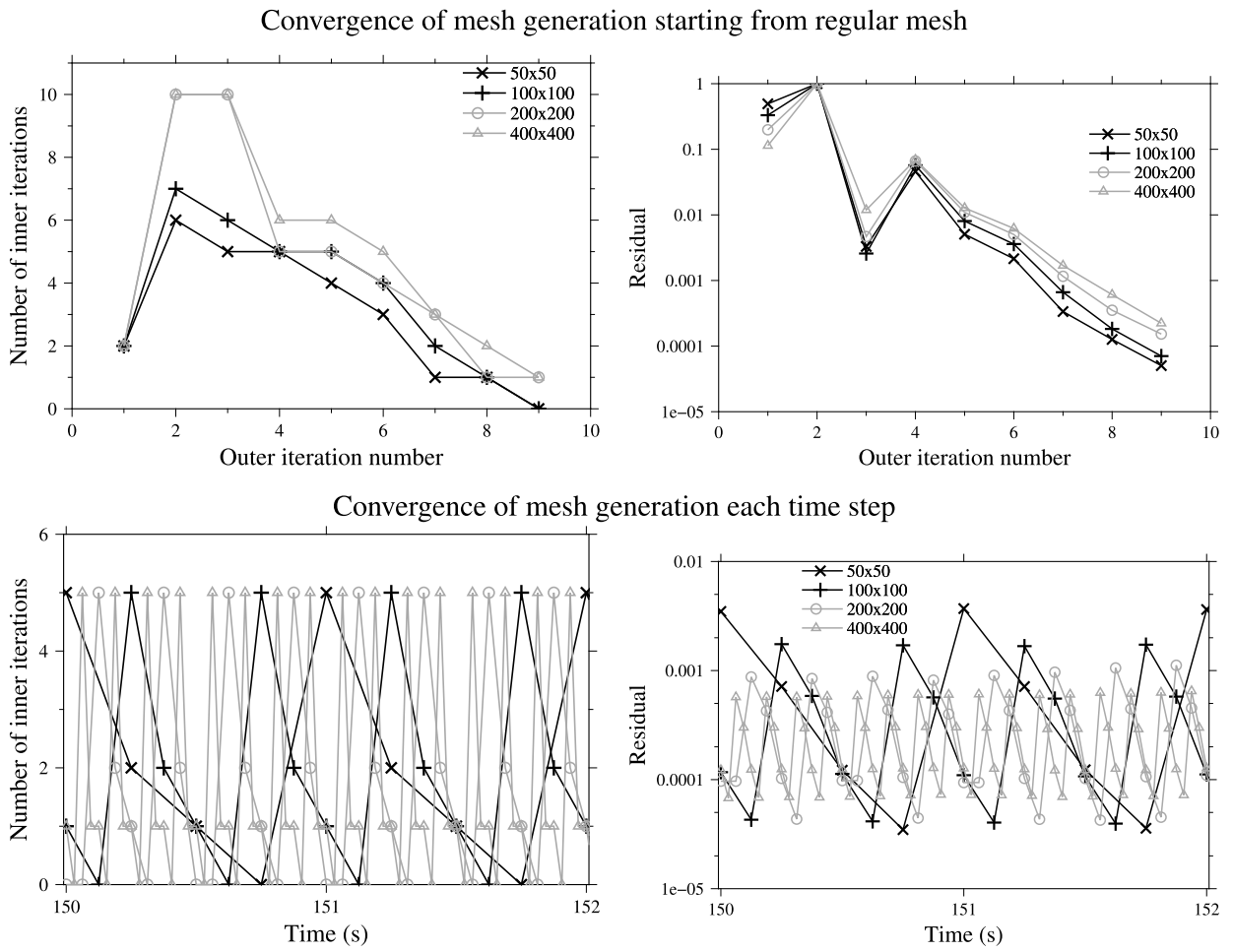

Fig. 13. Convergence of mesh generation for four different resolutions for advection over a hill and a valley. The top row shows convergence starting from a regular mesh. The bottom row shows convergence for each time step of the transient simulation. In the transient simulation there are four outer iterations, each consisting of one solution of the Poisson equation (32) initialised from the mesh at the previous time step. The left hand side shows the number of linear equation solvers per solution of the Poisson equation and the right hand side shows the initial residual before the linear equation solver is called. Before the simulation is started, a refined mesh is calculated from a uniform mesh and convergence is shown in the top row.

B.4. Results

Meshes are generated for the linear advection results using a cosine-shaped tracer in section 3.1 starting from initial uniform grids of 50×50, 100×100, 200×200 and 400×400 points in a plane of size 10 km by 10 km.

Before the advection simulation starts, an initial mesh is generated using the monitor function calculated from the analytic description of the initial conditions. 9 outer iterations are used. The residual of the Poisson equation solver and the number of iterations of the Poisson equation solver for each outer iteration are shown in the top row of Fig. 13 for all resolutions. Convergence is reasonably insensitive to resolution which is necessary for efficiency. Convergence in the first three iterations is noisy but then convergence proceeds exponentially (note the residuals are on a log-scale).

While solving the advection equation, the mesh is moved every time step. The same uniform, regular computational mesh is used to solve the Monge-Ampère equation each time step but the solution is initialised from the previous time step. Each time step, a maximum of 4 outer iterations of the Monge-Ampère Newton solver are allowed. The number of inner (linear equation solver) iterations and the initial residual for each solver are shown in the bottom row of Fig. 13. This shows that at most 5 inner iterations are needed and convergence is exponential between each outer-iteration per time step.

B.5. Further remarks

Section B.2 described a regularisation technique to ensure that the discretised Poisson equation (32) remains elliptic by artificially increasing the diagonal of the Poisson equation coefficient, P. This may raise concerns that we are arbitrarily

changing the problem that we are solving. However the regularisation is only very occasionally needed and is only ever needed during the first one or two outer iterations and so this regularisation never influences the final converged solution.

Browne et al. [3] compared this Newton solver with the parabolic method of Browne et al. [2] and with the fixed point iterations used by Weller et al. [19]. Convergence of the proposed Newton solver was far superior and free of arbitrary parameters. Browne et al. [3] also proposed a Newton solver that involved linearising the c/m term of (32). This lead to even faster but unreliable convergence and so is not used here.

References

[1] M.J. Berger, J. Oliger, Adaptive mesh refinement for hyperbolic partial differential equations, J. Comput. Phys. 53 (3) (1984) 484–512.
[2] P. Browne, C. Budd, C. Piccolo, M. Cullen, Fast three dimensional r-adaptive mesh redistribution, J. Comput. Phys. 275 (2014) 174–196.
[3] P. Browne, J. Prettyman, H. Weller, T. Pryer, J. van Lent, Nonlinear solution techniques for solving a Monge-Ampère equation for redistribution of a mesh, arXiv numerical analysis, arXiv:1609.09646, https://arxiv.org/abs/1609.09646, 2016.
[4] C.J. Budd, M. Cullen, E. Walsh, Monge–ampére based moving mesh methods for numerical weather prediction, with applications to the Eady problem, J. Comput. Phys. 236 (2013) 247–270.
[5] C.J. Budd, J. Williams, Moving mesh generation using the parabolic Monge–Ampère equation, SIAM J. Sci. Comput. 31 (5) (2009) 3438–3465.
[6] I. Demirdžić, M. Perić, Space conservation law in finite volume calculations of fluid flow, Int. J. Numer. Methods Fluids 8 (9) (1988) 1037–1050.
[7] G.S. Dietachmayer, K.K. Droegemeier, Application of continuous dynamic grid adaption techniques to meteorological modeling. Part I: basic formulation and accuracy, Mon. Weather Rev. 120 (8) (1992) 1675–1706.
[8] S.M. Guzik, X. Gao, L.D. Owen, P. McCorquodale, P. Colella, A freestream-preserving fourth-order finite-volume method in mapped coordinates with adaptive-mesh refinement, Comput. Fluids 123 (2015) 202–217.
[9] C. Hirt, A. Amsden, J. Cook, An arbitrary Lagrangian–Eulerian computing method for all flow speeds, J. Comput. Phys. 135 (2) (1997) 203–216.
[10] C. Kühnlein, P.K. Smolarkiewicz, A. Dörnbrack, Modelling atmospheric flows with adaptive moving meshes, J. Comput. Phys. 231 (7) (2012) 2741–2763.
[11] D. Long, J. Thuburn, Numerical wave propagation on non-uniform one-dimensional staggered grids, J. Comput. Phys. 230 (7) (2011) 2643–2659.
[12] A.T.T. McRae, C.J. Cotter, C.J. Budd, Optimal-transport-based mesh adaptivity on the plane and sphere using finite elements, SIAM J. Sci. Comput. 40 (2) (2018) A1121–A1148.
[13] A.M. Oberman, Wide stencil finite difference schemes for the elliptic Monge-Ampere equation and functions of the eigenvalues of the hessian, Discrete Contin. Dyn. Syst., Ser. B 10 (1) (2008) 221.
[14] P. Schwartz, J. Percelay, T. Ligocki, H. Johansen, D. Graves, D. Devendran, P. Colella, E. Ateljevich, High-accuracy embedded boundary grid generation using the divergence theorem, Commun. Appl. Math. Comput. Sci. 10 (1) (2015) 83–96.
[15] J. Shaw, H. Weller, J. Methven, T. Davies, Multidimensional method-of-lines transport for atmospheric flows over steep terrain using arbitrary meshes, J. Comput. Phys. 344 (2017) 86–107.
[16] W.C. Skamarock, J.B. Klemp, Adaptive grid refinement for two-dimensional and three-dimensional nonhydrostatic atmospheric flow, Mon. Weather Rev. 121 (3) (1993) 788–804.
[17] R. Vichnevetsky, Wave propagation and reflection in irregular grids for hyperbolic equations, Appl. Numer. Math. 3 (1987) 133–166.
[18] H. Weller, Predicting mesh density for adaptive modelling of the global atmosphere, Philos. Trans. R. Soc. Lond. A, Math. Phys. Eng. Sci. 367 (1907) (2009) 4523–4542.
[19] H. Weller, P. Browne, C. Budd, M. Cullen, Mesh adaptation on the sphere using optimal transport and the numerical solution of a Monge–Ampère type equation, J. Comput. Phys. 308 (2016) 102–123.