

Predicting Mesh Density for Adaptive Modelling of the Global Atmosphere

BY HILARY WELLER

NCAS Climate, Department of Meteorology, University of Reading, UK

The shallow water equations are solved using a mesh of polygons on the sphere which adapts infrequently to the predicted future solution. Infrequent mesh adaptation reduces the cost of adaptation and load balancing and will thus allow for more accurate mapping on adaptation.

We simulate the growth of a barotropically unstable jet adapting the mesh every 12 hours. Using an adaptation criterion based largely on the gradient of the vorticity leads to a mesh with around 20% of the cells of a uniform mesh which gives equivalent results. This is a similar proportion to previous studies of the same test case with mesh adaptation every one to twenty minutes.

The prediction of the mesh density involves solving the shallow water equations on a coarse mesh in advance of the locally refined mesh in order to estimate where features requiring higher resolution will grow, decay or move to. The adaptation criterion consists of two parts: that resolved on the coarse mesh and that which is not resolved and so is passively advected on the coarse mesh. This combination leads to a balance between resolving features controlled by the large scale dynamics and maintaining fine scale features.

Keywords: Finite volume, Voronoi, OpenFOAM, AtmosFOAM, Unstructured

Contents

1. Introduction	2
2. Solving the Shallow Water Equations on Polygons	3
3. Mesh to Mesh Interpolation and Mapping	4
(a) Interpolating from a Stencil of Cells onto a Point	5
(b) Ensuring Continuous and Bounded Interpolation	5
(c) Enforcing Local Conservation	6
(d) Variables to Interpolate	6
4. Adaptive Meshing with Polygons	7
(a) Voronoi Meshes based on a Mesh Density Function	7
(b) Predicting Mesh Density Ahead of Fluid Solution	8
5. Results of a Barotropically Unstable Jet	9
(a) Results using Fixed Meshes of Polygons	9
(b) Predictive Adaptive Meshing Results	11
(c) The Benefits of Predictive Adaptive Meshing	15
6. Summary and Conclusions	17

1. Introduction

Traditional models of the global atmosphere with fixed, uniform grids have proved immensely successful at weather and climate forecasting and their accuracy and efficiency are excellent. However, as computer power increases it may prove beneficial to increase resolution in some places more than others. For example, in order to resolve deep convection a prohibitive amount of additional resolution would be needed, but only over a small fraction of the globe. This motivates the need for adaptive mesh modelling of the atmosphere. Adaptive meshing has been under investigation for atmospheric modelling for some time [eg. Staniforth and Mitchell, 1978, Berger and Olinger, 1984, Skamarock et al., 1989, Dietachmayer and Droegemeier, 1992, Skamarock and Klemp, 1993, Fiedler and Trapp, 1993] but is still mostly research rather than operational [eg. Jablonowski et al., 2006, Läuter et al., 2007, St-Cyr et al., 2008], a noteworthy exception being Bacon et al. [2000]. If adaptive meshing is to be used to resolve deep convection, mesh density requirements must be predicted before convection would be likely to break out on the refined mesh. This paper tackles the issue of predicting mesh density requirements for simplified modelling of the global atmosphere, using the shallow water equations; a necessary precursor to predicting mesh density requirements for a more complete model of the global atmosphere.

Frequent re-meshing is usually necessary for adaptive mesh models of the global atmosphere because regions of high errors change frequently relative to the time step [eg Bacon et al., 2000, Jablonowski et al., 2006, Läuter et al., 2007, St-Cyr et al., 2008]. However re-meshing can be expensive and have a detrimental effect on the accuracy, reducing conservation of high moment properties of the flow and altering the partition between balanced and unbalanced flow. Re-meshing using unstructured meshes is particularly expensive for two reasons; ensuring conservation of even just mass requires calculating all the intersecting volumes between the old and new meshes and higher order schemes on unstructured meshes can be expensive to set up at the beginning of a fixed mesh run [eg Läuter et al., 2008, Weller et al., 2009] and hence expensive to re-initialise for each mesh change. However unstructured meshes may be desirable due to the improved accuracy from spatially smoothly varying refinement and the possibility of aligning the mesh with features. It may therefore be desirable to change the mesh relatively infrequently which necessitates predicting in advance regions of the globe which will require high resolution even if the flow in those regions may be quiescent at the time of the re-meshing.

Usually mesh density is determined based on instantaneous values of some local refinement criterion, a new mesh is created and sometimes high resolution can be expanded outwards so that adjacent cells do not differ in size by more than a factor of 2 [eg Jablonowski et al., 2006] or for unstructured meshes grading can be more gradual [eg Ringler et al., 2008]. This expanding outwards increases the number of time steps possible in between adaptation steps because it allows features to propagate away from their original region of fine mesh. But it should be possible to improve on this; running for even longer before re-meshing without expanding outwards in all directions as this will make the simulation unnecessarily expensive.

Low order adaptive mesh models of the shallow water equations have used relative vorticity, ζ [eg St-Cyr et al., 2008] as an adaptation criterion, or $\sqrt{\int_A \zeta^2 + \delta^2 dA}$

[Läuter et al., 2007] as an adaptation criterion, where dA is the cell area and δ is the divergence. In geophysical fluid dynamical applications of the shallow water equations the divergence is often small in comparison to the vorticity so these refinement criteria are similar in practice. However second order accurate discretisations of velocity and geopotential height should be able to represent high constant values of vorticity exactly on a coarse mesh; vorticity is proportional to the first spatial derivative of velocity and when close to geostrophic balance, velocity is proportional to the first derivative of height. This implies that vorticity is proportional to the second derivative of height. So if vorticity is constant then height will vary approximately quadratically which should be represented exactly using second order schemes. It may therefore be more appropriate to refine based on $|\nabla\zeta|$ or $|\nabla\sqrt{\zeta^2+\delta^2}|$ when using second order accurate schemes. As well as presenting a technique for re-meshing infrequently this paper also looks at the relative merits of refining based on $\sqrt{\zeta^2+\delta^2}$ and $|\nabla\sqrt{\zeta^2+\delta^2}|$.

Infrequent mesh adaptation has been achieved by Power et al. [2006] by solving a single large forward time step followed by single large adjoint backward step in order to estimate error norms for the flow's future movement. Hence remeshing was only necessary every 40 time steps. However the single large time step used assumes that changes are close to linear in between remeshing time steps. The technique presented here uses multiple time steps to determine the new mesh but on a coarse mesh to ensure that this process is inexpensive. However the error estimators are less advanced than in Power et al. [2006].

The model to solve the shallow water equations on fixed meshes of polygons is described in section 2 and various methods of mapping and interpolating between meshes are discussed in section 3. The mesh generator and the predictive adaptive meshing technique are described in section 4 and results demonstrating the value of the algorithm in simulating a barotropically unstable jet are given in section 5 with final conclusions drawn in section 6.

2. Solving the Shallow Water Equations on Polygons

The shallow water equations on a rotating sphere have been implemented using the computational fluid dynamics C++ library, OpenFOAM (www.opencfd.co.uk) to create AtmosFOAM as described in detail by Weller and Weller [2008] and Weller et al. [2009] and summarised here.

OpenFOAM solves transport equations in parallel using the finite volume method on 3-D polyhedral meshes in Cartesian coordinates. AtmosFOAM uses a spherical shell of polygonal cells embedded in 3-D space. It has not yet been run in parallel. The momentum and continuity equations are written in 3-D vector flux form:

$$\frac{\partial h\mathbf{U}}{\partial t} + \nabla \cdot (h\mathbf{U} \otimes \mathbf{U}) = -2\boldsymbol{\Omega} \times h\mathbf{U} - gh\nabla(h + h_0) \quad (2.1)$$

$$\frac{\partial h}{\partial t} + \nabla \cdot (h\mathbf{U}) = 0, \quad (2.2)$$

where \mathbf{U} is the 3-D velocity vector in global Cartesian co-ordinates relative to the rotating frame, h is the height of the fluid surface above the solid surface, h_0 is the height of the solid surface above a spherical reference height, $\boldsymbol{\Omega}$ is the angular velocity vector of the sphere ($= (0, 0, 7.292 \times 10^{-5} s^{-1})$), g is the scalar acceleration due to gravity and $\nabla \cdot$ and ∇ are the divergence and gradient operators in 3-D

global Cartesian space. Because of the use of global 3-D Cartesian coordinates, a Lagrange multiplier should be added to the momentum equation to constrain the motion to follow the sphere [as suggested by Côté, 1988]; instead any radial component of the momentum is removed during each time step ($h\mathbf{U} \rightarrow h\mathbf{U} - (h\mathbf{U} \cdot \hat{\mathbf{r}})\hat{\mathbf{r}}$ where $\hat{\mathbf{r}}$ is the unit normal vector in the radial direction).

The prognostic variables are the three components of the cell-average momentum, $h\mathbf{U}$ and the cell average height, h . The mass flux between cells, ϕ , is advanced from the interpolated momentum at the previous time step by solving the momentum equation as described by Rhie and Chow [1983], removing the grid scale oscillations of the A-grid [Arakawa and Lamb, 1977]. So ϕ is partially a prognostic variable.

Here is an overview of the solution algorithm:

1. The cell centred momentum equation is linearised about the solution from the previous iteration and each component solved implicitly. This allows time steps slightly longer than the CFL restriction based on the flow.
2. The momentum equation is also discretised on the cell faces and is combined with the continuity equation to form a modified Helmholtz equation to predict the height and the face fluxes, ϕ . This allows time steps much longer than that restricted by gravity waves.
3. The implicit time discretisation is two time-level Crank-Nicholson with off centering of 0.55 (0.5 being centred).
4. Centred, linear differencing is used to construct the global matrices with higher order explicit deferred corrections. This leads to very sparse, diagonally dominant matrices which are efficiently solved with iterative solvers.
5. Equations are solved twice per time step with higher order discretisation, non-linear and Coriolis terms updated for the second iteration to improve accuracy and convergence. The second iteration effectively makes the algorithm fully implicit in terms of the accuracy.
6. The higher order corrections fit either a two-dimensional quadratic or a polynomial with a few of the additional terms from a cubic. An upwind biased stencil of cells is used for discretising the divergence term of the momentum equation using the partial cubic. A centred stencil of cells is used for discretising all other terms using the quadratic. The 2-D quadratic and partial cubics are:

$$a_0 + a_1x + a_2y + a_3x^2 + a_4xy + a_5y^2 \quad (2.3)$$

$$a_0 + a_1x + a_2y + a_3x^2 + a_4xy + a_5y^2 + a_6x^3 + a_7xy^2 \quad (2.4)$$

and examples of the upwind biased and centred stencils are shown in figure 1.

3. Mesh to Mesh Interpolation and Mapping

This section provides some comments on how solutions should be mapped (conservatively) from one mesh to another as well as how mesh to mesh interpolation

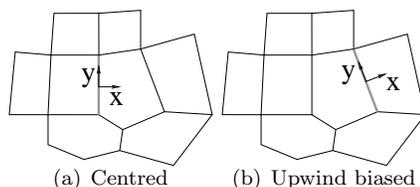


Figure 1. The 2D stencils of cells for interpolating from cell centres onto the grey face.

(non-conservatively) is achieved for the solutions presented in this paper (section a). The mesh to mesh interpolation implemented is not optimal but is sufficient to demonstrate the validity of the mesh density prediction technique presented here. Comparisons with other mesh to mesh mappings (sections b to d) are the subject of future work.

(a) Interpolating from a Stencil of Cells onto a Point

For the results presented in section 5b, a two-dimensional quadratic (equation 2.3) is used to interpolate from stencils of cell centres in the old mesh onto each new cell centre. The stencils in the old mesh are found starting from the cell(s) which contain(s) the new cell centre and expanding outwards until the stencil satisfies two criteria:

1. The stencil must contain more cells than unknowns in the quadratic (*i.e.* 6). This ensures that the matrix to be inverted to find the coefficients in the quadratic is over- rather than under-specified and is solved with a least squares fit.
2. The stencil of old cells must cover all the vertices of the new cell. This ensures that all cells in the old mesh are used in the mesh to mesh interpolation, even if the old mesh is much finer than the new.

The advantage of this technique is that it gives quadratic accuracy on arbitrarily unstructured meshes. However there are some disadvantages:

1. It is not locally conservative as interpolation is from old to new cell centres rather than from old to new areas or volumes.
2. Because the 2D quadratic is only fit in a least squares sense, the interpolation is discontinuous between adjacent stencils in the old mesh.
3. The interpolation is not guaranteed to be bounded because it uses a quadratic.

The advantages and disadvantages of this technique are the same as using bi-cubic spline interpolation or radial basis function interpolation as described in chapter 4.5 of Behrens [2006]. The disadvantages may be overcome as described in some of the following subsections.

(b) Ensuring Continuous and Bounded Interpolation

The most straightforward way of creating a continuous interpolation would be to treat the dual mesh of triangles (or quadrilaterals for meshes of quadrilaterals) as a

P1 continuous finite element mesh with the cell average values at polygon centroids becoming the point values at the nodes of the triangles. Linear interpolation to new cell centres within a triangle can then be done which would ensure both continuous and bounded interpolation. But this would not be conservative.

(c) *Enforcing Local Conservation*

Local conservation requires the calculation of all of the intersecting areas between the cells of the old and new mesh which is considerably more effort on an unstructured mesh [Iske and Kaser, 2004, and fig 2] in comparison to the trivial job that this is on a block-structured mesh. Efficient algorithms are available for calculating the necessary intersections and areas [eg O'Rourke et al., 1982]. If fields are interpolated non-conservatively from the old cell centres to the cell centres of the intersections of the old and new meshes then a conservation correction can be added to ensure that the new intersecting areas contain exactly the same amount as the old cell which they cover. Local conservation corrections like this (and like the mass-packet based algorithm described by Behrens [2006]) reduce the formal order of accuracy.

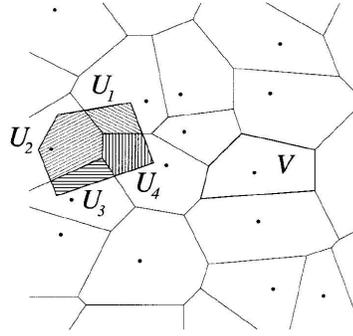


Figure 2. Figure 4 from Iske and Kaser [2004] showing new cell U decomposed into four parts which intersect with the old cells V for interpolating from old (V) to new (U) cells. Iske and Kaser [2004] used this algorithm for conservative semi-Lagrangian but it is assumed here to be for arbitrary mesh to mesh mapping.

(d) *Variables to Interpolate*

If mass (geopotential height) and momentum are mapped from old to new mesh as described in sections b and c then mass and momentum will be conserved locally but there will be no guarantees that the partition between geostrophic and ageostrophic velocity will be preserved. Alternatively, for a two-dimensional shallow-water model, vorticity, divergence and mass could be mapped conservatively from old to new meshes. This interpolation would effectively be higher order than interpolating mass and momentum because vorticity and divergence are derivatives of velocity. This mapping would lead to conservation of vorticity rather than momentum which may have advantages and would lead to improved preservation of geostrophic balance under adaptation. However retrieving components of

velocity from vorticity and divergence requires the solution of the Poisson equation which comes at additional cost.

4. Adaptive Meshing with Polygons

Section a describes how to create a Voronoi mesh of polygons from a mesh density function. The novel technique for defining this mesh density function based on the predicted requirements until the next re-meshing time is described in section b.

Currently AtmosFOAM runs on a fixed mesh and separate OpenFOAM executables have been created for generating new meshes based on the mesh density function and mapping solutions between meshes. The executables are all called from an outer time loop in a script.

(a) Voronoi Meshes based on a Mesh Density Function

In order to generate Voronoi meshes of polygons in which the distances between cell centres conform approximately to a required mesh density function, a set of points on the sphere are Delaunay triangulated using the Computational Geometry Algorithms Library [CGAL, <http://www.cgal.org>]. Each triangle edge is then modelled as a critically damped spring with un-stretched length equal to the desired mesh density function and the mesh of springs is then relaxed, following Tomita et al. [2002]. In order to converge quickly to a mesh with density close to the given function, points are also added to and removed from the triangulation in regions where they are too closely or too sparsely packed following Weller et al. [2009]. The Voronoi dual is then used as the mesh for AtmosFOAM (eg figure 3).

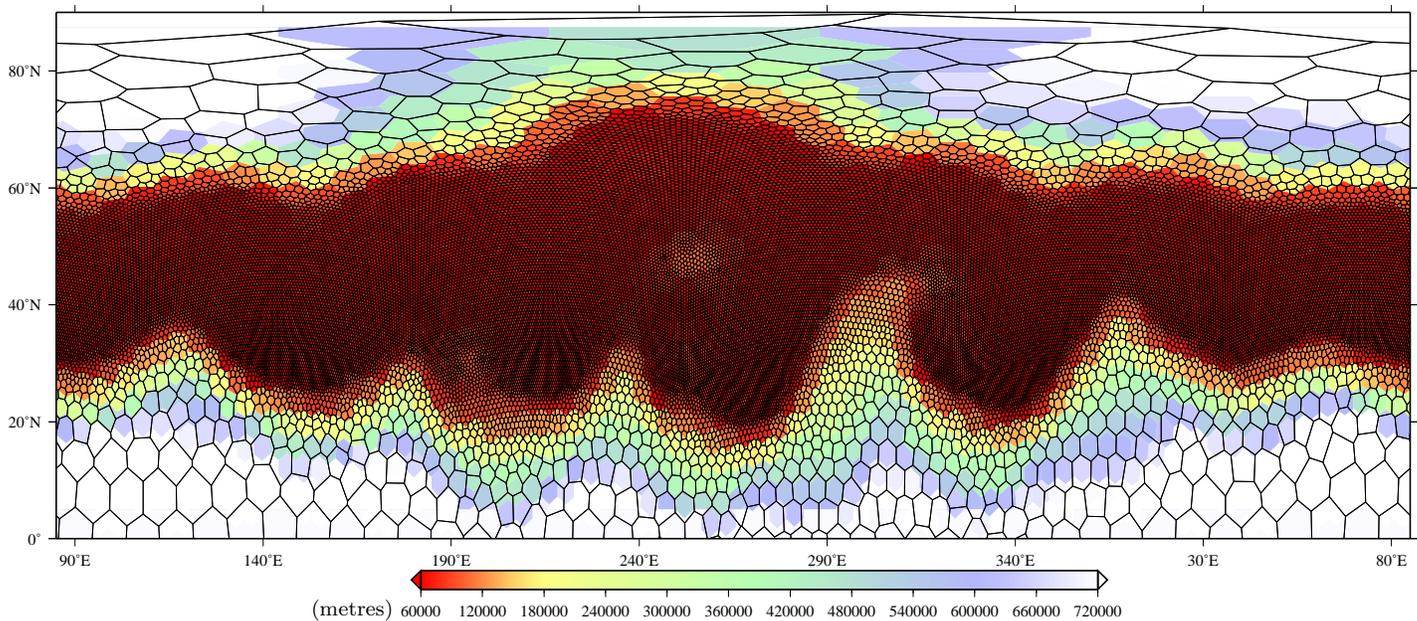


Figure 3. The locally refined mesh generated from the density function shown by colours. The density function is calculated and displayed on a uniform coarse mesh.

(b) *Predicting Mesh Density Ahead of Fluid Solution*

It now remains to define the most important and novel part of this adaptive meshing algorithm; how we define the mesh density function in order to generate a mesh for the next simulation of the global atmosphere until the next re-meshing time. The time between re-meshings is denoted T , from T_0 to T_1 and the required mesh density function, $\frac{1}{\delta x}$. The full shallow water equations on a coarse mesh are solved between T_0 and T_1 in advance, in order to predict where $\frac{1}{\delta x}$ may need to be high. The initial conditions on the coarse mesh are interpolated from the fine mesh at T_0 . As well as keeping track of where refinement criteria are high on the coarse mesh based on the flow that is resolved on the coarse mesh, a measure of the features which are unresolved on the coarse mesh is also advected with the coarse mesh solution so that the new mesh can be generated to resolve these features.

In section 5 we will show results using two refinement criteria; $R = \eta = \sqrt{\zeta^2 + \delta^2}$ (where ζ is the local horizontal component of the relative vorticity, $\zeta = \hat{\mathbf{r}} \cdot (\nabla \times \mathbf{U})$ where $\hat{\mathbf{r}}$ is the radial direction and δ is the divergence) and for $R = |\nabla \eta|$. The part of the required mesh density based on the flow which is resolved on the coarse mesh, $\frac{1}{\delta x_r}$ (where subscript r indicates resolved on the coarse mesh) is given by:

$$\frac{1}{\delta x_r} = \frac{1}{\delta x_{\max}} + \frac{\max\left(R_{rc}, \max_{T_0 \rightarrow T_1} R_r\right)}{R_{rc}} \left(\frac{1}{\delta x_{\min}} - \frac{1}{\delta x_{\max}} \right) \quad (4.1)$$

where R_r is either η or $|\nabla \eta|$ resolved on the coarse mesh, $\max_{T_0 \rightarrow T_1}$ takes the maximum value over every time step between times T_0 and T_1 for every coarse mesh cell, R_{rc} is the critical value for the resolved component of R , δx_{\max} is the maximum resolution to be used where R is zero and δx_{\min} is the minimum resolution to be used where R is at or above the critical value.

The part of the required mesh density based on the flow which may not be resolved on the coarse mesh is calculated by first calculating R on the old fine mesh at time T_0 and injecting into each coarse mesh cell the maximum fine mesh value over all the fine mesh cells whose centre lies within the coarse mesh. For example the maximum is taken over all the cells shown by dots in figure 4 and this value is given to the coarse mesh cell which contains all the dots. These injected values are then advected passively with the flow which is solved on the coarse mesh and the maximum value seen in each cell over time T is stored as $\max_{T_0 \rightarrow T_1} R_u$ where subscript u denotes unresolved. The part of the required mesh density based on the flow which may not be resolved on the coarse mesh, $\frac{1}{\delta x_u}$ is given by:

$$\frac{1}{\delta x_u} = \frac{1}{\delta x_{\max}} + \frac{\max\left(R_{uc}, \max_{T_0 \rightarrow T_1} R_u\right)}{R_{uc}} \left(\frac{1}{\delta x_{\min}} - \frac{1}{\delta x_{\max}} \right) \quad (4.2)$$

The required mesh density based on both the resolved and unresolved fields is then calculated as:

$$\frac{1}{\delta x} = \max\left(\frac{1}{\delta x_r}, \frac{1}{\delta x_u}\right) \quad (4.3)$$

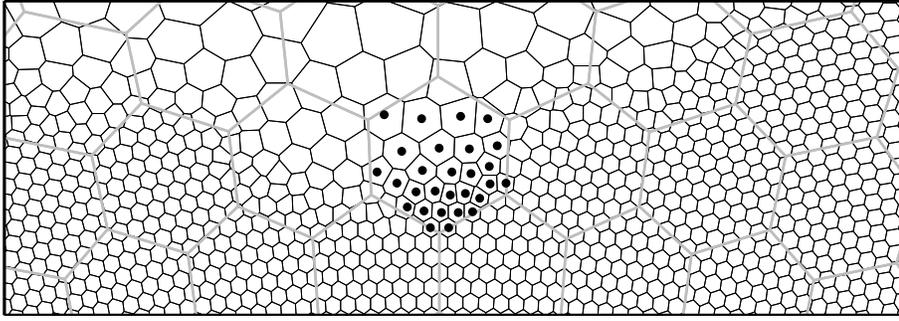


Figure 4. Coarse and fine mesh cells. The unresolved component of the flow is estimated by taking the maximum value of the refinement criteria, R , in the fine mesh over all the cells whose centre lies within a coarse mesh cell (as shown by dots). This maximum value is then injected into the coarse mesh cell

and the field is smoothed without increasing resolution anywhere by taking, for each cell, the maximum between $\frac{1}{\delta x}$ and the average value of all the neighbours of the cell.

From the mesh density function, $\frac{1}{\delta x}$, a new mesh is created as described in section a, the solution is interpolated from the old to new meshes at time T_0 as described in section 3 and the atmosphere is simulated on the new mesh until time T_1 as described in section 2.

5. Results of a Barotropically Unstable Jet

The barotropically unstable jet test case of Galewsky et al. [2004] is a challenging test case for both unstructured meshes and mesh adaptivity; firstly because barotropic instability is particularly sensitive to numerical errors with spurious waves generated where geostrophic balance is lost due to imbalanced truncation errors and secondly because there is a cascade of length scales from planetary to sub-grid scale so at some point small scale features should no longer be resolved.

(a) Results using Fixed Meshes of Polygons

The AtmosFOAM shallow water model has been validated by Weller et al. [2009] but some additional results of the barotropically unstable jet on fixed polygonal and reduced latitude-longitude meshes are shown here.

The relative vorticity of the barotropically unstable jet after six days on six different AtmosFOAM meshes are shown in figure 5 and are compared with the spectral reference solution of Galewsky et al. [2004]. The fixed time steps (given in figure 5) lead to maximum flow CFL numbers of about 0.4 and maximum gravity wave CFL numbers of about 0.9. All the resolutions of reduced latitude-longitude meshes appear to be very similar to the reference solution but without the spectral ringing. The upwind biased differencing of the non-linear advection term (the divergence term in the flux form momentum equation) means that the model is slightly diffusive and once features cascade to the smallest scales they are diffused.

The solutions on reduced latitude-longitude meshes are also compared with solutions on quasi-uniform meshes of polygons (hexagonal icosahedral) with various

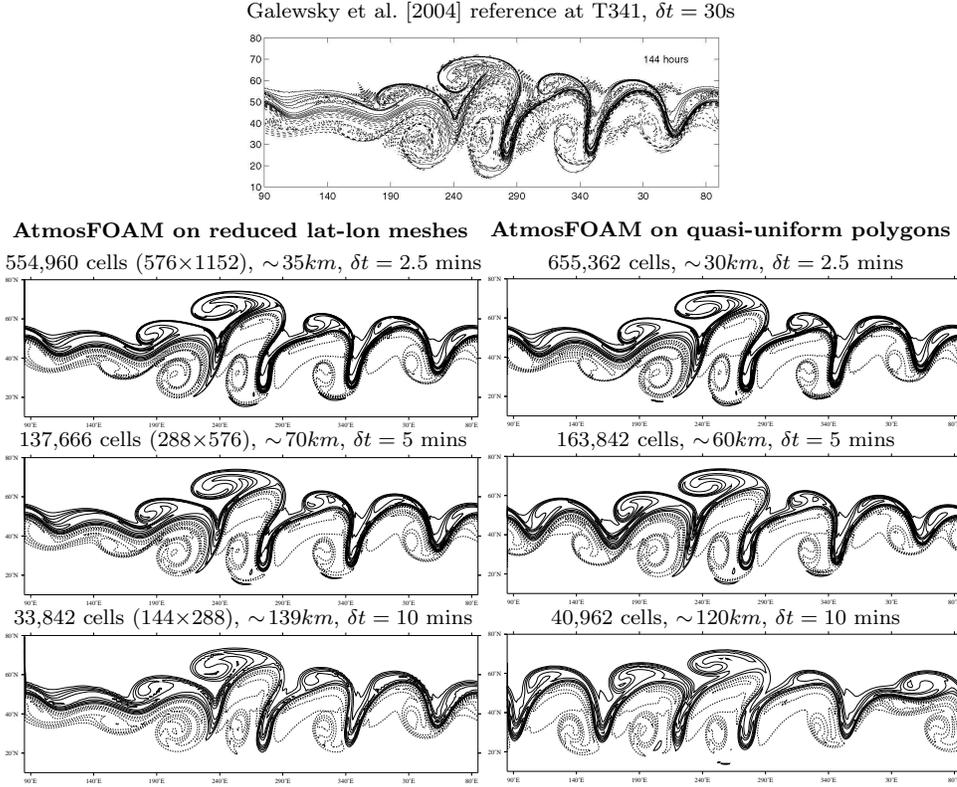


Figure 5. Relative vorticity of the barotropically unstable jet after 6 days. Contours every $2 \times 10^{-5} s^{-1}$. Results on six fixed quasi-uniform AtmosFOAM meshes are compared with the spectral reference solution of Galewsky et al. [2004].

similar resolutions. The reduced latitude-longitude meshes do not generate spurious waves (as do the coarser meshes of polygons) because the jet is perfectly aligned with the unperturbed flow before the waves grow and so barotropic instability is not generated by unbalanced truncation errors on the aligned latitude-longitude meshes. However when this case is run on rotated reduced latitude-longitude meshes, results are similar or worse than using the meshes of polygons [Weller et al., 2009].

The growth of errors with time on the different AtmosFOAM meshes is shown in figure 6 – the root mean square normalised volumetric mean height errors (the ℓ_2 error norm) for the AtmosFOAM cases in comparison to the finest reduced latitude-longitude mesh. There is an initial jump in the errors due to the coarser resolution of the initial gravity wave adjustment. Between half and two days the errors of the coarser models grow very little. After 2.5 days the errors on polygonal meshes grow more quickly due to the barotropic waves triggered by the non-alignment of the meshes.

When creating adapting meshes of polygons in section b, a finest resolution of $60km$ is used as this appears to represent the barotropic wave with reasonable accuracy (despite slightly enhanced wave growth at around $115^\circ E$ at 6 days in fig 5) while being cheaper to compute than the mesh of $30km$ polygons.

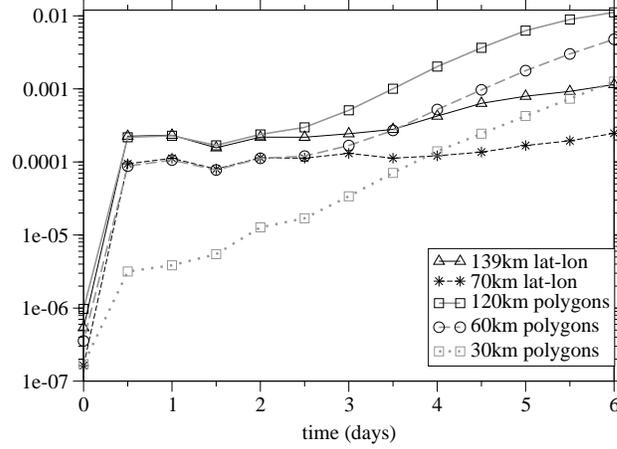


Figure 6. Root mean square normalised volumetric mean height errors (ℓ_2) of the barotropically unstable jet on fixed meshes relative to the finest reduced latitude-longitude mesh (576×1152).

(b) *Predictive Adaptive Meshing Results*

The barotropically unstable jet is simulated with a mesh of polygons adapting every $T = 12$ hours with a fixed time step of 5 minutes and a finest resolution of 60km , simulating the 12 hours in advance using a coarse mesh and adapting based on various critical values of the two refinement criteria, $R = \eta = \sqrt{\zeta^2 + \delta^2}$ and $R = |\nabla\eta|$ as defined in table 1. The coarse mesh consists of 2,562 quasi-

R	R_{rc}	R_{uc}	shorthand
η	$2 \times 10^{-5} \text{ s}^{-1}$		$\eta_c = (2, 2) \times 10^{-5} \text{ s}^{-1}$
η	$4 \times 10^{-5} \text{ s}^{-1}$		$\eta_c = (4, 4) \times 10^{-5} \text{ s}^{-1}$
η	$8 \times 10^{-5} \text{ s}^{-1}$		$\eta_c = (8, 8) \times 10^{-5} \text{ s}^{-1}$
η	$2 \times 10^{-5} \text{ s}^{-1}$	$8 \times 10^{-5} \text{ s}^{-1}$	$\eta_c = (2, 8) \times 10^{-5} \text{ s}^{-1}$
$ \nabla\eta $	$1 \times 10^{-10} \text{ s}^{-1} \text{ m}^{-1}$		$ \nabla\eta _c = (1, 1) \times 10^{-10} \text{ s}^{-1} \text{ m}^{-1}$
$ \nabla\eta $	$2 \times 10^{-10} \text{ s}^{-1} \text{ m}^{-1}$		$ \nabla\eta _c = (2, 2) \times 10^{-10} \text{ s}^{-1} \text{ m}^{-1}$
$ \nabla\eta $	$4 \times 10^{-10} \text{ s}^{-1} \text{ m}^{-1}$		$ \nabla\eta _c = (4, 4) \times 10^{-10} \text{ s}^{-1} \text{ m}^{-1}$
$ \nabla\eta $	$1 \times 10^{-10} \text{ s}^{-1} \text{ m}^{-1}$	$4 \times 10^{-10} \text{ s}^{-1} \text{ m}^{-1}$	$ \nabla\eta _c = (1, 4) \times 10^{-10} \text{ s}^{-1} \text{ m}^{-1}$

Table 1. Various refinement criteria and their critical values used in the adaptive mesh simulations of the atmosphere

uniform polygons (2,550 hexagons and 12 pentagons) with an approximate distance of 480km in between cell centres. For each 12 hour period, 8 time steps of 90 minutes are simulated on the coarse mesh which leads to a maximum flow CFL number of about 0.9 and a maximum gravity wave CFL number of about 2 and 144 time steps of 5 minutes are simulated on the adapted mesh which leads to a maximum flow CLF number of about 0.4 and a maximum gravity wave CFL number of about 0.9. As an example of the coarse mesh results, the simulated vorticity which is resolved

on the coarse mesh and the injected, unresolved vorticity which is purely advected on the coarse mesh are shown in figure 7 from initialisation at 5.5 days until 6 days. These simulations are evidently not as accurate as the fine mesh results and

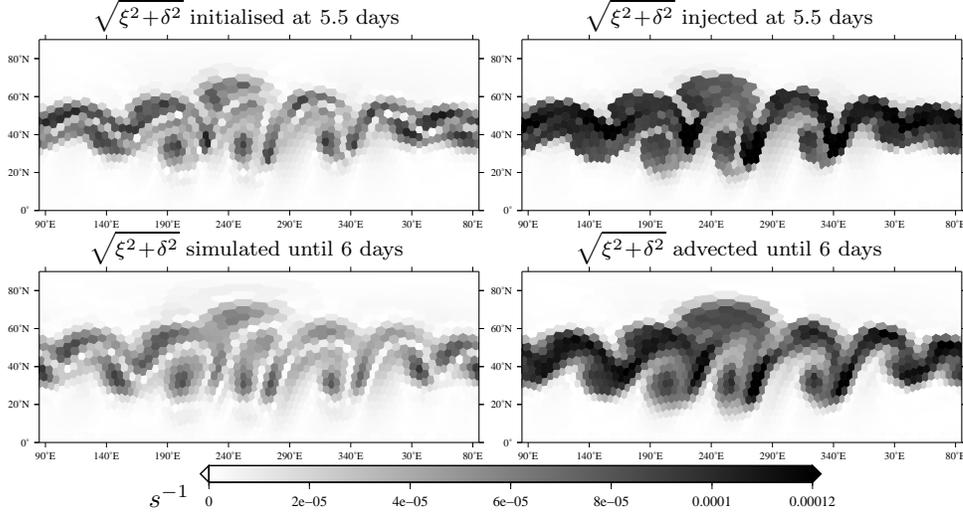


Figure 7. Solutions on the coarse mesh initialised at 5.5 days and simulated until 6 days using the refinement criteria, $R = \eta_c = (4, 4) \times 10^{-5} s^{-1}$

so cannot predict precisely where resolution will be needed. But the crests of the waves are expanding polewards and the troughs in between the waves are expanding to the south. So increased resolution will be created to the north and resolution will be decreased in the troughs.

The vorticity fields at day 6 using all of the refinement criteria and critical values are shown in figure 8 which also shows contours of the mesh resolution used to simulate between days 5.5 and 6. All of the refinement criteria used give qualitatively similar results to the fixed mesh solution in figure 8 apart from $|\nabla\eta|_c = (4, 4) \times 10^{-10} s^{-1} m^{-1}$ which leads to a mesh which is too coarse. The height errors relative to the mesh of fixed 60km polygons at six days are shown in figure 9. (Comparison with results from fixed 60km polygons is not comparison with a reference solution but it is comparison with the best achievable solution using an adaptive mesh with the same minimum cell size.) This confirms the excessive errors using the refinement criterion $|\nabla\eta|_c = (4, 4) \times 10^{-10} s^{-1} m^{-1}$ due to the triggering of spurious barotropic instability where the mesh is non-uniform and too coarse.

Figure 9 shows that both the $|\nabla\eta|$ criteria and $|\eta|$ criteria give similar errors for the area covered by the finest resolution. This can be understood with reference to figure 8 which shows that in most places, high magnitudes of vorticity are co-located with high vorticity gradients. The exception to this is in the wave troughs on the southern flank of the jet at $\sim 300^\circ E$ and $\sim 0^\circ$ where near uniform values of non-zero vorticity are present. This then leads to the $|\nabla\eta|$ criteria diagnosing lower mesh density here than the η criteria. This leads to the $|\nabla\eta|$ refinement criteria giving meshes with fewer cells for the same errors as confirmed in figure 10 which shows the number of cells as a function of time and the volumetric mean root mean square error relative to the fixed uniform mesh. The errors of the fixed uniform

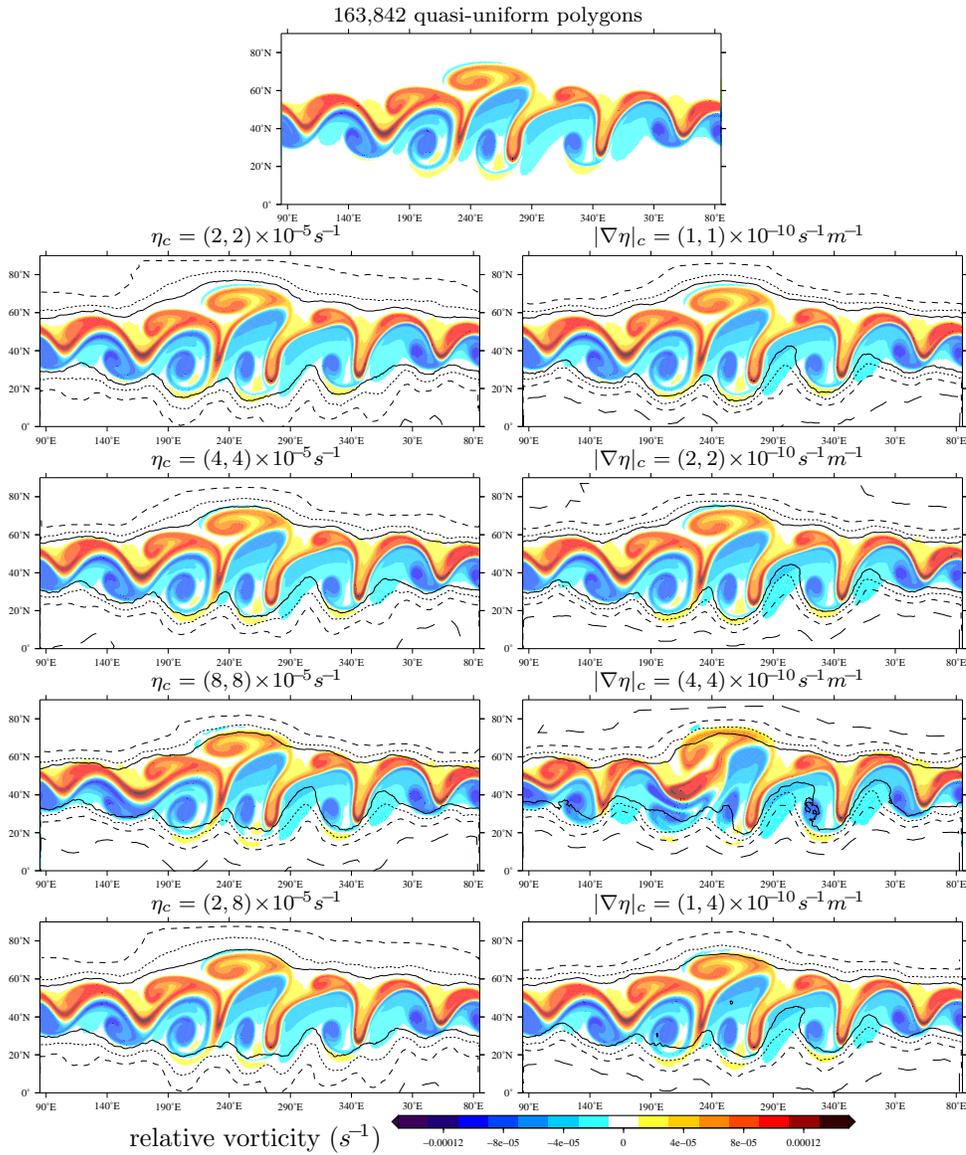


Figure 8. Relative vorticity of the barotropically unstable jet of after 6 days on various adaptive meshes. Contours show resolution, solid $80km$, small dashed $160km$, medium dashed, $320km$, long dashed $640km$. Labels above plots show the refinement criteria. Quasi-uniform mesh has approximate resolution $60km$.

mesh of $60km$ polygons is also plotted relative to the reduced latitude-longitude mesh of $35km$ resolution.

From figures 9 and 10, resolution depending on both the resolved and unresolved components is clearly important. The large differences in errors between using, for example, $|\nabla\eta|_c = (4, 4) \times 10^{-10} s^{-1} m^{-1}$ and $(1, 4) \times 10^{-10} s^{-1} m^{-1}$ clearly show the benefit of using a sufficiently low criterion for the resolved component. But if the

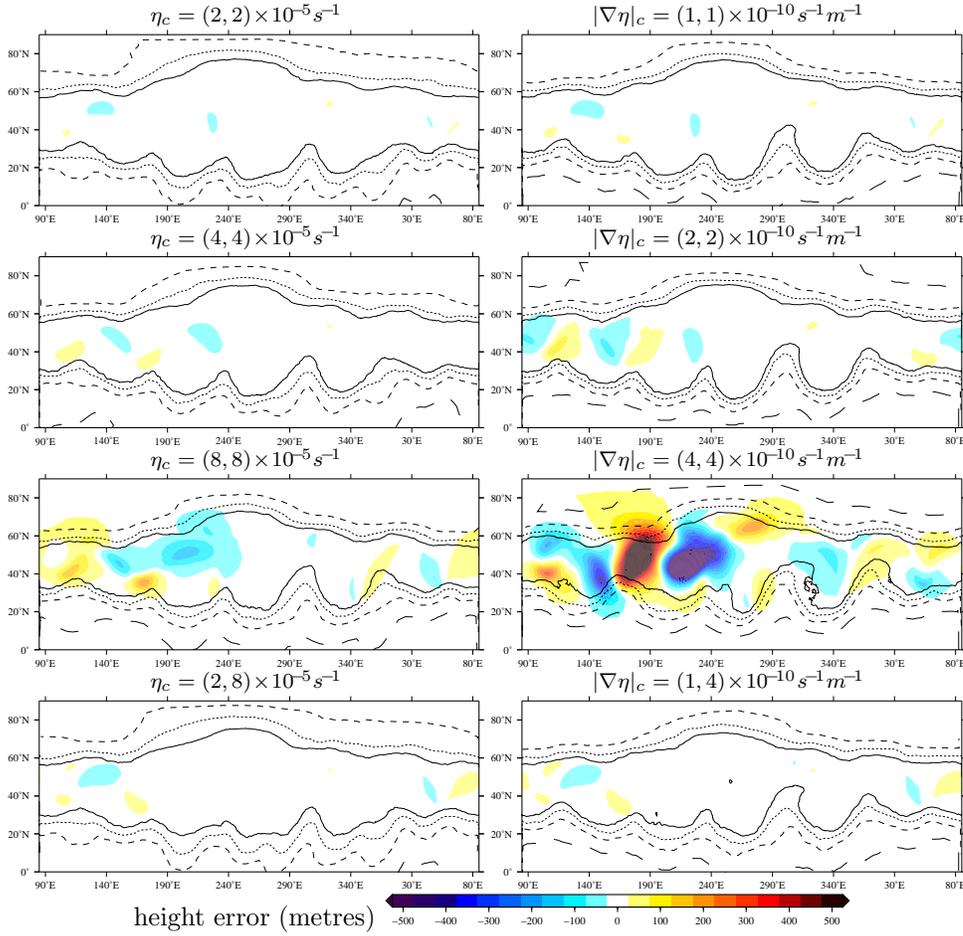


Figure 9. Height errors of the adaptive mesh solutions relative to mesh of 163,842 quasi-uniform polygons of approximate resolution 60km after 6 days on various adaptive meshes. Contours show resolution, solid 80km , small dashed 160km , medium dashed, 320km , long dashed 640km . Labels above plots show the refinement criteria.

critical value for the unresolved component is too low (eg $(1, 1) \times 10^{-10} \text{s}^{-1} \text{m}^{-1}$ then high mesh density is predicted to be needed over too much area.

The initial jump in error for the adaptive mesh runs in figure 10 is larger than the fixed quasi-uniform run with the same finest resolution because the initial gravity wave adjustment phase which leads to this initial error consists of gravity waves propagating globally which are not resolved by the adaptive mesh runs. The errors then remain constant for a few days and the type of mesh refinement dictates when the errors start to grow. The $|\nabla\eta|_c = (4, 4) \times 10^{-10} \text{s}^{-1} \text{m}^{-1}$ criterion which leads to very few cells throughout the run leads to the earliest increase in errors and the largest errors throughout. The errors in comparison to the fixed uniform mesh run are larger than the errors of the fixed uniform mesh run and we can conclude that this adaptation criterion is inadequate. The next cheapest adaptation criterion is $\eta_c = (8, 8) \times 10^{-5} \text{s}^{-1}$. This is much larger than the value of critical relative vorticity

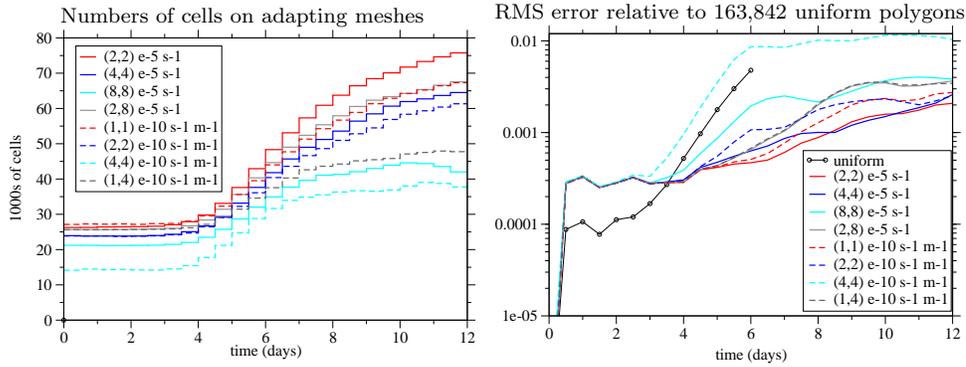


Figure 10. Number of cells and root mean square volumetric mean height error relative to mesh of 163,842 quasi-uniform polygons of approximate resolution $60km$. The RMS error labelled “uniform” is relative to the finest reduced latitude-longitude mesh.

of $3 \times 10^{-5} s^{-1}$ used by St-Cyr et al. [2008] and the errors start to increase after about four days for this simulation, half a day earlier than the other simulations.

An adaptation criterion that appears to be an attractive compromise between accuracy and efficiency from figures 8, 9 and 10 is $|\nabla\eta|_c = (1, 4) \times 10^{-10} s^{-1} m^{-1}$ as the resolution is initially high and hence errors are initially low and errors do not begin to grow until about 4.5 days. After 5 days the errors using this criterion grow slightly more quickly than some of the other simulations but the number of cells used grows very little, unlike the other criteria. This asymmetric criterion resolves the initial features which are resolved on the coarse mesh prediction mesh well because it uses the low value of $|\nabla\eta|_{rc} = 1 \times 10^{-10} s^{-1} m^{-1}$ but does not resolve so well the finer scale features which develop later in the simulation which are not well represented on the coarse mesh because it uses $|\nabla\eta|_{uc} = 4 \times 10^{-10} s^{-1} m^{-1}$. This could be an advantage for adaptive mesh modelling because if features continually cascade to smaller scales until they are dissipated by molecular viscosity, mesh adaptation criteria will put resolution everywhere unless criteria such as these which consider more strongly the larger scales of the flow are used.

Using the favoured adaptation criterion of $|\nabla\eta|_c = (1, 4) \times 10^{-10} s^{-1} m^{-1}$, the vorticity before and after re-meshing at 5.5 days and the vorticity at 6 days are shown in figure 11 and the mesh density is contoured. This shows the vorticity of the jet filling up the region of finest mesh at 5.5 days before re-meshing. After re-meshing new regions of fine mesh are created to the north in preparation for the waves to grow northwards as is seen at day 6. Resolution is decreased in some of the troughs on the southern flank of the jet due to the propagation of the waves out of the fine mesh.

(c) The Benefits of Predictive Adaptive Meshing

The cost of predicting mesh density requirements on a coarse mesh (2,562 cells in this case) is tiny in comparison to the refined mesh solutions which use tens of thousands of cells. The cost of generating new unstructured meshes and mapping solutions accurately between meshes has not been estimated as the mesh generation has not been optimised and conservative mapping has not yet been done. This cost is likely to be considerable which is the motivation for the infrequent adaptation.

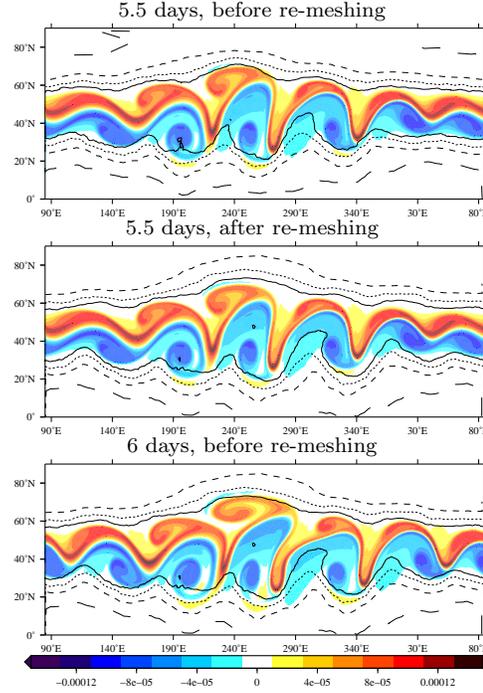


Figure 11. The mesh density (contours) before and after re-meshing at 5.5 days using the $|\nabla\eta|_c = (1, 4) \times 10^{-10} s^{-1} m^{-1}$ criterion and the relative vorticity (colours) at 5.5 days before and after interpolation and the vorticity simulated on the new mesh until 6 days

However if re-meshing and mapping is only done every 12 hours in comparison to a time step of 5 minutes then this cost becomes insignificant.

The $|\nabla\eta|_c = (1, 4) \times 10^{-10} s^{-1} m^{-1}$ adaptation criterion leads to meshes with 26 to 48 thousand cells, 16% to 30% of the 163,844 cells of the quasi-uniform mesh of polygons with the same finest resolution while the errors are very similar. This cost saving is not enough to justify the use of adaptive meshing for weather or climate forecasting alone. But for more realistic situations there could be greater benefits if high resolution features are more sparsely distributed, for example fronts, tropical cyclones and deep convection.

St-Cyr et al. [2008] compared finite volume and spectral element adaptive mesh models using this test case and achieved accuracy very similar to their uniform meshes using 21% of the adapting finite volumes or 23% of the adapting spectral elements at day 6. This compares with AtmosFOAM which uses 21% of the cells at day 6 using the $|\nabla\eta|_c = (1, 4) \times 10^{-10} s^{-1} m^{-1}$ adaptation criterion. However re-meshing is done only every 12 hours using AtmosFOAM whereas St-Cyr et al. [2008] adapt every time step (50 seconds) using their finite volume model or every 20 minutes (400 time steps) using their spectral element model. Läuter et al. [2007] simulated a slower jet with a shallow water model adapting every time step and found a 22% saving in number of cells at day 15 in comparison to a uniformly fine mesh. The mesh density prediction gives a very impressive result; meshes with the same density of cells are generated but re-meshing only needs to be done every 12 hours for this case as opposed to every 50 seconds or 20 minutes when adapting

based on instantaneous criteria. This means that much more effort can be put into the mesh to mesh mapping to ensure greatest accuracy as this does not need to be done frequently.

6. Summary and Conclusions

A method of solving equations of motion on adapting meshes of polygons on the sphere is presented. The aim is to calculate in advance where high resolution is likely to be needed so that re-meshing need only be done infrequently. This reduces the cost of re-meshing by up to the reduction in frequency, for example re-meshing every 12 hours rather than every 20 minutes will reduce the cost of re-meshing by up to a factor of 36. In practice the gain will be less than this because of the small additional cost of the mesh density prediction step.

The prediction of mesh density is done by mapping the latest conditions onto a coarse mesh and thus solving the equations of motion cheaply until the next re-meshing time. Refinement criteria are accumulated on the coarse mesh over this simulation so that a locally refined mesh can be generated in order to simulate this period again more accurately, starting from the latest conditions mapped directly from old to new refined meshes.

Two refinement criteria, R_r and R_u , are accumulated simultaneously on each coarse mesh in order to determine where fine resolution will be needed (where subscript r refers to the component of the flow which is resolved on the coarse mesh and subscript u the component which may not be well resolved on the coarse mesh). Two formulations for R have been tested, $R = \eta = \sqrt{\zeta^2 + \delta^2}$ where ζ is the relative vorticity and δ is the divergence and $R = |\nabla\eta|$. The resolved component, R_r is simply calculated from the solution on the coarse mesh at every coarse mesh time step. R_u is calculated initially for the latest time on the locally refined mesh and, for each coarse mesh cell, the maximum value over each fine mesh cell whose centre lies within the coarse mesh cell is taken and this value is injected into the coarse mesh cell. These injected values are then advected passively on the coarse mesh and refinement criteria are again accumulated at each time step. A combination of the two refinement criteria determines the mesh density for generating a new locally refined mesh for simulating the flow accurately.

It was expected that mesh density prediction together with infrequent adaptation would lead to higher simulation costs at the expense of reduced re-meshing and mapping costs due to the generation of meshes with significantly more cells than adaptation based on instantaneous values. However this has not been found to be the case for the test case studied: meshes have a similar density of cells in comparison to adaptive mesh models which re-mesh much more frequently [Läuter et al., 2007, St-Cyr et al., 2008]. The prediction of the mesh density requirements means that high resolution mesh is created in regions before it is needed which has great accuracy advantages – in particular it means that these simulations are accurate even using simple, non-conservative and non-bounded mesh to mesh interpolation.

Weller et al. [2009] have shown that there may be accuracy advantages in using unstructured meshes to simulate the global atmosphere due to the possibility of using spatially smoothly varying refinement rather than abrupt block structured refinement with 2:1 cell splitting. If this is used with occasional re-meshing and mesh density prediction as described here then adaptive unstructured meshes may

also prove efficient and higher order numerical schemes which can be expensive to set up on unstructured meshes [eg Läuter et al., 2008] can also be used cost-effectively.

Extending the mesh density prediction technique to three dimensions leads to some interesting questions: if mesh adaptation is to be uniform in the vertical direction, should simulations to predict mesh density requirements be two dimensional representations of the 3D atmosphere? This would ensure very low cost of the mesh density prediction. It may work for some barotropic features of the mid-latitudes but probably not for baroclinic states or tropical disturbances. So the mesh density prediction will probably need to be 3D in order to predict a 3D mesh with 2D refinement. But, given the value of current weather and climate predictions which use coarse resolution, coarse resolution will probably be effective for mesh density prediction.

Adaptive meshing could be particularly beneficial for resolving deep tropical convection, one of the weakest sub-grid scale parameterizations in current climate models. However it is not clear how to diagnose where to place resolution before convection breaks out, as convection can only break out with realistic timing where resolution is already high. This problem may be surmountable if current convection parameterizations are re-tuned to predict mesh density requirements rather than using them to predict convection. This could fit in easily alongside the mesh density prediction method presented here.

Finally, the infrequent re-meshing made possible by mesh density prediction will make domain decomposition and parallelisation more efficient as cells will need to be moved between processors less frequently during the load balancing necessary on re-meshing.

To conclude, infrequent mesh adaptation and mesh density prediction will make adaptive modelling of the atmosphere more efficient and will allow for more accurate mapping on adaptation. This comes at a small additional cost of the mesh prediction step and potentially some additional cost of having larger areas with high mesh resolution, although this has not been found for the test case presented.

References

- A. Arakawa and V. Lamb. Computational design of the basic dynamical processes of the UCLA general circulation model. *Methods in Computational Physics*, 17: 173–265, 1977.
- D. Bacon, N. Ahmad, Z. Boybeyi, T. Dunn, M. Hall, P. Lee, R. Sarma, M. Turner, K. Waight, S. Young, and J. Zack. A dynamically adapting weather and dispersion model: The Operational Multiscale Environment Model with Grid Adaptivity (OMEGA). *Mon. Wea. Rev.*, 128(7):2044–2076, 2000.
- J. Behrens. *Adaptive Atmospheric Modelling*. Number 54 in Lecture Notes in Computational Science and Engineering. Springer, 2006.
- M. Berger and J. Olinger. Adaptive Mesh Refinement for Hyperbolic Partial Differential Equations. *J. Comput. Phys.*, 53:484–512, 1984.
- CGAL. Computational Geometry Algorithms Library, <http://www.cgal.org>.

- J. Côté. A Lagrange multiplier approach for the metric terms of semi-Lagrangian models on the sphere. *Quart. J. Roy. Meteor. Soc.*, 114:1347–1352, 1988.
- G. Dietachmayer and K. Droegemeier. Application of continuous dynamic grid adaption techniques to meteorological modeling. Part I: Basic formulation and accuracy. *Mon. Wea. Rev.*, 120(8):1675–1706, 1992.
- B. Fiedler and R. J. Trapp. A fast dynamic grid adaption scheme for meteorological flows. *Mon. Wea. Rev.*, 121(10):2879–2888, 1993.
- J. Galewsky, R. Scott, and L. Polvani. An initial-value problem for testing numerical models of the global shallow-water equations. *Tellus*, 56A(5):429–440, 2004.
- A. Iske and M. Kaser. Conservative semi-Lagrangian advection on adaptive unstructured meshes. *Numerical Methods for Partial Differential Equations*, 20(3):388–411, 2004.
- C. Jablonowski, M. Herzog, J. Penner, R. Oehmke, Q. Stout, B. van Leer, and K. Powell. Block-structured adaptive grids on the sphere: Advection experiments. *Mon. Wea. Rev.*, 134(12):3691–3713, 2006.
- M. Läuter, D. Handorf, N. Rakowsky, J. Behrens, S. Frickenhaus, M. Best, D. K., and W. Hiller. A parallel adaptive barotropic model of the atmosphere. *J. Comput. Phys.*, 223(2):609–628, 2007.
- M. Läuter, F. Giraldo, D. Handorf, and K. Dethloff. A discontinuous galerkin method for the shallow water equations in spherical triangular coordinates. *J. Comput. Phys.*, 227:10226–10242, 2008.
- J. O’Rourke, C.-B. Chien, T. Olson, and D. Naddor. A new linear algorithm for intersecting convex polygons. *Computer graphics and image processing*, 19:384–391, 1982.
- P. Power, M. Piggott, F. Fang, G. Gorman, C. Pain, D. Marshall, A. Goddard, and I. Navon. Adjoint goal-based error norms for adaptive mesh ocean modelling. *Ocean Modelling*, 15:3–38, 2006.
- C. Rhie and W. Chow. Numerical study of the turbulent-flow past an airfoil with trailing edge separation. *AIAA Journal*, 21(11):1525–1532, 1983.
- T. Ringler, L. Ju, and M. Gunzburger. A multiresolution method for climate system modeling: applications of spherical centroidal Voronoi tessellations. *Ocean Dynamics*, Online first, 2008.
- W. Skamarock and J. Klemp. Adaptive grid refinement for two-dimensional and three-dimensional nonhydrostatic atmospheric flow. *Mon. Wea. Rev.*, 121(3):788–804, 1993.
- W. Skamarock, J. Olinger, and R. Street. Adaptive grid refinement for numerical weather prediction. *J. Comput. Phys.*, 80(1):27–60, 1989.
- A. St-Cyr, C. Jablonowski, J. Dennis, H. Tufo, and S. Thomas. A comparison of two shallow-water models with nonconforming adaptive grids. *Mon. Wea. Rev.*, 136:1898–1922, June 2008.

- A. Staniforth and H. Mitchell. Variable-resolution finite-element technique for regional forecasting with primitive equations. *Mon. Wea. Rev.*, 106(4):439–447, 1978.
- H. Tomita, M. Satoh, and K. Goto. An optimization of the icosahedral grid modified by spring dynamics. *J. Comput. Phys.*, 183(1):307–331, 2002.
- H. Weller and H. Weller. A high-order arbitrarily unstructured finite-volume model of the global atmosphere: tests solving the shallow-water equations. *Int. J. Numer. Meth. Fluids*, 56(8):1589–1596, 2008.
- H. Weller, H. Weller, and A. Fournier. Voronoi, Delaunay and block structured mesh refinement for solution of the shallow water equations on the sphere. *Mon. Wea. Rev.*, submitted, 2009.