Adaptive Mesh Refinement for Three-Dimensional Off-Line Tracer Advection over the Sphere

M. E. Hubbard
School of Computing, University of Leeds

1 Introduction

Adaptive Mesh Refinement (AMR) is a commonly-used technique in the field of Computational Fluid Dynamics, which improves the efficiency of existing numerical models by distributing the nodes of the computational mesh in such a way that high mesh resolution is provided only where it is necessary to achieve the level of accuracy required of the approximation. So far, AMR has not been incorporated fully into operational meteorological models, though many partial attempts have been made. Often, this is due to the numerical schemes generally used within the community being less suited for use on variable resolution meshes than those widely used in other fields. Consequently, the work presented in this paper combines the AMR with a standard conservative, flux-based, finite volume scheme and applies the resulting algorithm to the approximation of global atmospheric flows based on a regular latitude/longitude/height mesh.

2 Adaptive Mesh Refinement

The Adaptive Mesh Refinement (AMR) algorithm used is essentially that of Berger and Oliger [3] and Berger and Colella [2], which provides a straightforward process by which the resolution of structured Cartesian meshes can be adapted locally to efficiently give the desired accuracy when approximating partial differential equations.

The method employs a hierarchical system of overlaid, properly nested, embedded grids, as illustrated in two dimensions by the solid lines in Fig. 1, and these grids are continually updated to ensure that the regions of high resolution follow the moving flow features. All communication between meshes is carried out via the transfer of solution information into layers of dummy cells, such as those illustrated by dotted lines in Fig. 1, supplying boundary conditions for integrations on individual meshes. For flexibility, the relative level of refinement between two grid levels can take any positive integer value (even 1) and need not be the same in any of the three coordinate directions. The adaptation is controlled by a very simple monitor function,

$$\xi_{ijk} = \max(|\psi_{i+1jk} - \psi_{i-1jk}|, |\psi_{ij+1j} - \psi_{ij-1k}|, |\psi_{ijk+1} - \psi_{ijk-1}|)\,, \tag{2.1}$$

where ψ is the advected variable and (i, j, k) index the structured meshes. The new meshes are formed using the clustering algorithm of Bell *et al.* [1], and adaptive time-stepping is used to improve efficiency, and a flux correction is applied at mesh boundaries to ensure conservation.

On the sphere, the meshes are regularly spaced in spherical polar coordinates, and a single coarse mesh covers the whole of the globe. Beyond this, the application of the adaptive mesh refinement algorithm remains essentially unchanged. The only modifications required are

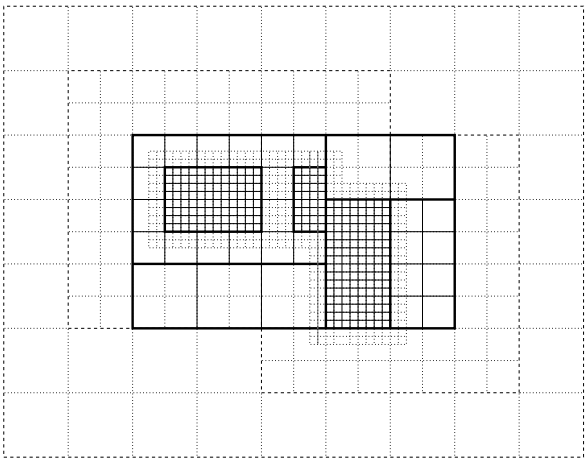

Figure 1: A two-dimensional adapted grid with the dummy cells superimposed.

- The transfer of solution information from fine meshes to coarse meshes is carried out through a volume weighted averaging procedure to ensure conservation.
- The transfer of solution information from coarse meshes to fine meshes (used to fill both the dummy cells surrounding the mesh boundaries and the newly created mesh structure) is still conservative when piecewise constant interpolation is used, but no longer when a MUSCL-type reconstruction [6] is applied to the solution. Here a correction is applied in which the MUSCL reconstruction is scaled to ensure that the sum of the 'masses' contained within each fine cell is the same as the mass within the corresponding coarse cell. The scaled reconstruction is no longer guaranteed to be monotonic, but it does prohibit any change of sign.
- The communication between meshes through layers of dummy cells which span either of the poles must take into account the reversal of the coordinate system across the singularity. As a result the velocity field swaps sign in the dummy cells and any filling of strips of cells is carried out in reverse. Zonal boundaries require periodic conditions to fill the dummy cells.

When the calculations are forced with externally supplied data this is projected on to the grids using trilinear interpolation in space and linear interpolation in time.

3 The Scalar Advection Equation

The equation being approximated here is the multidimensional scalar advection equation

$$\psi_t + \vec{u} \cdot \vec{\nabla} \psi \;=\; 0\,, \tag{3.1}$$

in which ψ represents the advected scalar quantity, such as a chemical mixing ratio, and $\vec{u} = (u, v, w)^{\mathrm{T}}$ is the velocity vector.

Equation (3.1) is not itself a conservation law (except in the special case when $\vec{\nabla} \cdot \vec{u} \equiv 0$), but it may be rewritten

$$\psi_t + \vec{\nabla} \cdot (\psi \vec{u}) \;=\; \psi \, \vec{\nabla} \cdot \vec{u}\,. \tag{3.2}$$

The left hand side is in conservation form, and is discretised using a flux-based finite volume technique, whilst the right hand side acts as a forcing term, and is discretised in a manner

which maintains conservation in the special case when the discrete representation of the velocity field is divergence-free.

The approximation is constructed using a forward Euler discretisation of the time derivative, along with dimensional splitting which can be carried out on the regular structured meshes created by the AMR. A cell centred finite volume scheme is used to approximate the flux terms. This is combined with a consistent approximation of the velocity divergence term, designed to maintained a constant flow field ($\psi = K$) indefinitely, whatever the velocity field. Conservation is assured when the discrete form of the divergence of the velocity field is zero. The resulting scheme, in three dimensions, takes the form

$$\begin{aligned}
\psi^{\star}_{ijk} &= \psi^{n}_{ijk} - \frac{\Delta t}{V_{ijk}} \times \{\, [A\,(\psi u)^{*}\,]^{n}_{i+1/2jk} - [A\,(\psi u)^{*}\,]^{n}_{i-1/2jk} \\
&\qquad + \overline{\psi}^{n}_{ijk}\,[A\,u^{*}\,]^{n}_{i+1/2jk} - \overline{\psi}^{n}_{ijk}\,[A\,u^{*}\,]^{n}_{i-1/2jk}\,\} \\
\psi^{\star\star}_{ijk} &= \psi^{\star}_{ijk} - \frac{\Delta t}{V_{ijk}} \times \{\, [A\,(\psi v)^{*}\,]^{\star}_{ij+1/2k} - [A\,(\psi v)^{*}\,]^{\star}_{ij-1/2k} \\
&\qquad + \overline{\psi}^{n}_{ijk}\,[A\,v^{*}\,]^{n}_{ij+1/2k} - \overline{\psi}^{n}_{ijk}\,[A\,v^{*}\,]^{n}_{ij-1/2k}\,\} \\
\psi^{n+1}_{ijk} &= \psi^{\star\star}_{ijk} - \frac{\Delta t}{V_{ijk}} \times \{\, [A\,(\psi w)^{*}\,]^{\star\star}_{ijk+1/2} - [A\,(\psi w)^{*}\,]^{\star\star}_{ijk-1/2} \\
&\qquad + \overline{\psi}^{n}_{ijk}\,[A\,w^{*}\,]^{n}_{ijk+1/2} - \overline{\psi}^{n}_{ijk}\,[A\,w^{*}\,]^{n}_{ijk-1/2}\,\}\,, \qquad (3.3)
\end{aligned}$$

where $\cdot^{\star}$ and $\cdot^{\star\star}$ represent the intermediate states of the update procedure, and A and V are, respectively, cell face areas and cell volumes.

The numerical fluxes are approximated using the WAF approach, originally proposed by Toro [5] for the modelling of hyperbolic conservation laws. Since the WAF approach is being applied here to the advection equation rather than a conservation law, the numerical flux is not used in precisely the usual form. The numerical flux is assumed to take the form $f^{*} = (\psi u)^{*} = \psi^{*} u^{*}$, so the WAF flux becomes

$$(\psi u)^{*}_{i+1/2} \;=\; \frac{1}{2}(1+\phi_{i+1/2})u_{i+1/2}\psi_{i} + \frac{1}{2}(1-\phi_{i+1/2})u_{i+1/2}\psi_{i+1}\,, \qquad (3.4)$$

where $u_{i+1/2} = (u_i + u_{i+1})/2$ and ϕ is an 'amplified' CFL number which incorporates the flux limiter (superbee is used here). The expressions for the numerical fluxes, (3.4) and those related to the other two space dimensions, are substituted directly into (3.3) to give the WAF-type scheme used here.

In spherical polar coordinates, and for a regular latitude(θ)/longitude(λ)/height(r) structured grid the volumes and face areas are calculated exactly, with constant mesh spacings $\Delta\lambda$, $\Delta\theta$ and Δr. The WAF-type flux given by (3.4) is used without any modification, except that the local dual cell CFL number used in the calculation of ϕ is now taken to be

$$\nu_{i+1/2} \;=\; u_{i+1/2}\Delta t \frac{2A_{i+1/2}}{V_i + V_{i+1}}\,, \qquad (3.5)$$

to account for the variation in the cell sizes over the computational domain. The cell centre representation of the solution ensures that no information is stored at the poles and the singularities there cause no problems: only a small amount of additional bookkeeping is required to account for the reversal of the coordinate system for cross-polar communication. The only requirement is that there be an even number of coarse grid cells (in the context of AMR) in the zonal direction.

Grid	Equivalent resolution	Errors l_1	l_2	l_∞	Time (s)
64×32	64×32	0.5828	0.4567	0.4590	25
64×32 ($\times 2$)	128×64	0.1325	0.1355	0.1870	78
128×64	128×64	0.1296	0.1320	0.1794	370
64×32 ($\times 2 \times 3$)	384×192	0.0661	0.0616	0.0795	1526
128×64 ($\times 3$)	384×192	0.0611	0.0559	0.0677	2074
384×192	384×192	0.0466	0.0438	0.0607	32807

Table 1: Error measures for the solid body rotation of a cosine bell after one revolution ($\alpha = \pi/2$) using a superbee limited WAF scheme with varying degrees of adaptation.

4 Numerical Experiments

The first problem used here to test the code is essentially the standard two-dimensional solid body rotation test case of Williamson *et al.* [7] in which a cosine bell shaped profile is advected, without distortion, around the sphere at an angle $\pi/2$ to the equator, *i.e.* over the poles. The bell radius R is set here to $7\pi/64$. All numerical experiments were run with a maximum Courant number of 0.9.

The AMR requires specification of certain control parameters. In this case the condition $\xi \leq 0.1$ (using (2.1) as the monitor ξ) was used for the cell flagging, along with one additional layer of buffer cells, and the remeshing was carried out every time-step. The cell clustering parameter η is taken to be 0.9 so a high proportion of cells in the fine grids have been flagged.

Table 1 shows the savings which are made in terms of cpu time. The timings are obtained on a SUN ULTRA 10 workstation, but it should be emphasised that no additional acceleration techniques have been used in the approximation to counteract the effects of the converging grid at the poles (*cf.* Table 2). There are two main reasons for the dramatic improvements seen: the reduction in the number of computational cells on which the integration is carried out, and the relative increase in the stable time-step for the period of the adaptive calculation when the fine grid does not cover either of the poles.

The effect of applying multiple sweeps in the zonal direction is illustrated in Table 2, which shows a series of results obtained on a 384×192 grid. The level of reduction referred to in the first column indicates the number of times the grid would have been 'reduced' by a factor of 2 in the zonal direction to gain an equivalent increase in the global time-step. The number of sweeps increases towards the poles by a factor of 2 at equispaced intervals of $\cos\theta$, except for the case denoted 'optimal' where the number of sweeps can take any positive integer value, not just a power of 2, and is calculated separately for every layer of cells.

The method has also been applied to the idealised cyclogenesis problem of Nair *et al.* [4] with similarly encouraging results. The increase in speed obtained by the AMR is slightly less dramatic because a larger proportion of the mesh has been refined, but a speed-up of a factor of 3 is still simple to obtain, with no loss of accuracy and no spurious oscillations in the solution at the steep interface, which is captured very sharply.

In three dimensions, the method has been applied to solid body rotation around the sphere with an additional vertical velocity component. The initial solution profile, when viewed in a cylindrical projection, takes the form of a slotted sphere and is transported over the poles in the manner of the cosine bell of of Williamson *et al.* [7], but with an additional sinusoidal oscillation in the vertical with one quarter the period of the revolution around the sphere.

Equivalent level of 'reduction'	Errors l_1	l_2	l_∞	Time (s)	Time-step (s)
0	0.0466	0.0438	0.0607	32807	20.0
1	0.0464	0.0437	0.0605	17485	39.8
2	0.0461	0.0435	0.0604	9106	79.5
3	0.0453	0.0430	0.0593	4988	159.1
4	0.0442	0.0421	0.0587	2872	318.1
5	0.0505	0.0461	0.0683	1701	636.2
6	0.0822	0.0723	0.0749	1192	1215.1
7	0.0828	0.0729	0.0758	1720	1215.1
'Optimal'	0.0772	0.0689	0.0758	684	1215.1

Table 2: Error measures for the solid body rotation of a cosine bell ($\alpha = \pi/2$) after one revolution using a superbee limited WAF scheme with varying numbers of zonal sweeps on a single 384×192 mesh.

More precisely, an initial sphere is defined by

$$\psi \;=\; \begin{cases} 1.0 & \text{if } \; d < \pi/6 \\ 0.0 & \text{otherwise} \end{cases} \tag{4.1}$$

where $d = [(\lambda - \lambda_c)^2 + (\theta - \theta_c)^2 + (h - h_c)^2]^{1/2}$, in which $(\lambda_c, \theta_c, h_c) = (\pi/2, 0.0, h_{\max}/2)$. A slot is cut out of the sphere by supplementing this with the condition

$$\psi \;=\; 0.0 \quad \text{if} \quad r \geq r_c - \pi/18 \quad \text{and} \quad \theta_c + \pi/18 \geq \theta \geq \theta_c - \pi/18\,, \tag{4.2}$$

where r is given by

$$r \;=\; \cos^{-1}[\sin\theta_c \, \sin\theta + \cos\theta_c \, \cos\theta \, \cos(\lambda - \lambda_c)]\,, \tag{4.3}$$

the great circle distance between (λ, θ) and the bell centre, initially taken as $(\lambda_c, \theta_c) = (\pi/2, 0.0)$. This is also the exact solution after any integer number of revolutions around the sphere. The velocity field is

$$\begin{aligned} u &= u_0(\cos\alpha \, \cos\theta + \sin\alpha \, \cos\lambda \, \sin\theta) \\ v &= -u_0 \, \sin\alpha \, \sin\lambda \\ w &= 0.00002 \cos(\pi t/129600)\,, \end{aligned}$$

where $u_0 = 38.61\text{ms}^{-1}$ and $\alpha = \pi/2$, giving the solid body rotation a period of 12 days and the vertical oscillation a period of 3 days (t is given in seconds). $h_{\max}$ is the magnitude of the computational domain in the vertical direction.

The solutions shown in Fig. 2 have been obtained on a $64 \times 32 \times 32$ coarse grid, covering the domain $[0 : 2\pi, -\pi/2 : \pi/2, 0 : \pi]$, with a single level of grid refinement (by a factor of 4 in each direction). The adapted grid varies between approximately 180000 and 430000 cells, while the equivalent fine grid would contain 4194304 cells. It works out that the adaptive run takes about 1/13 of the cpu time of the fine grid run and uses roughly 1/3 of the memory. Also note that there is no distortion in the profile as it crosses the polar singularities.

The combined WAF/AMR algorithm, has also been extensively tested against the evolution of observed atmospheric data, for which it is driven by externally supplied velocity fields and initial conditions. The results will be presented in future publications but they

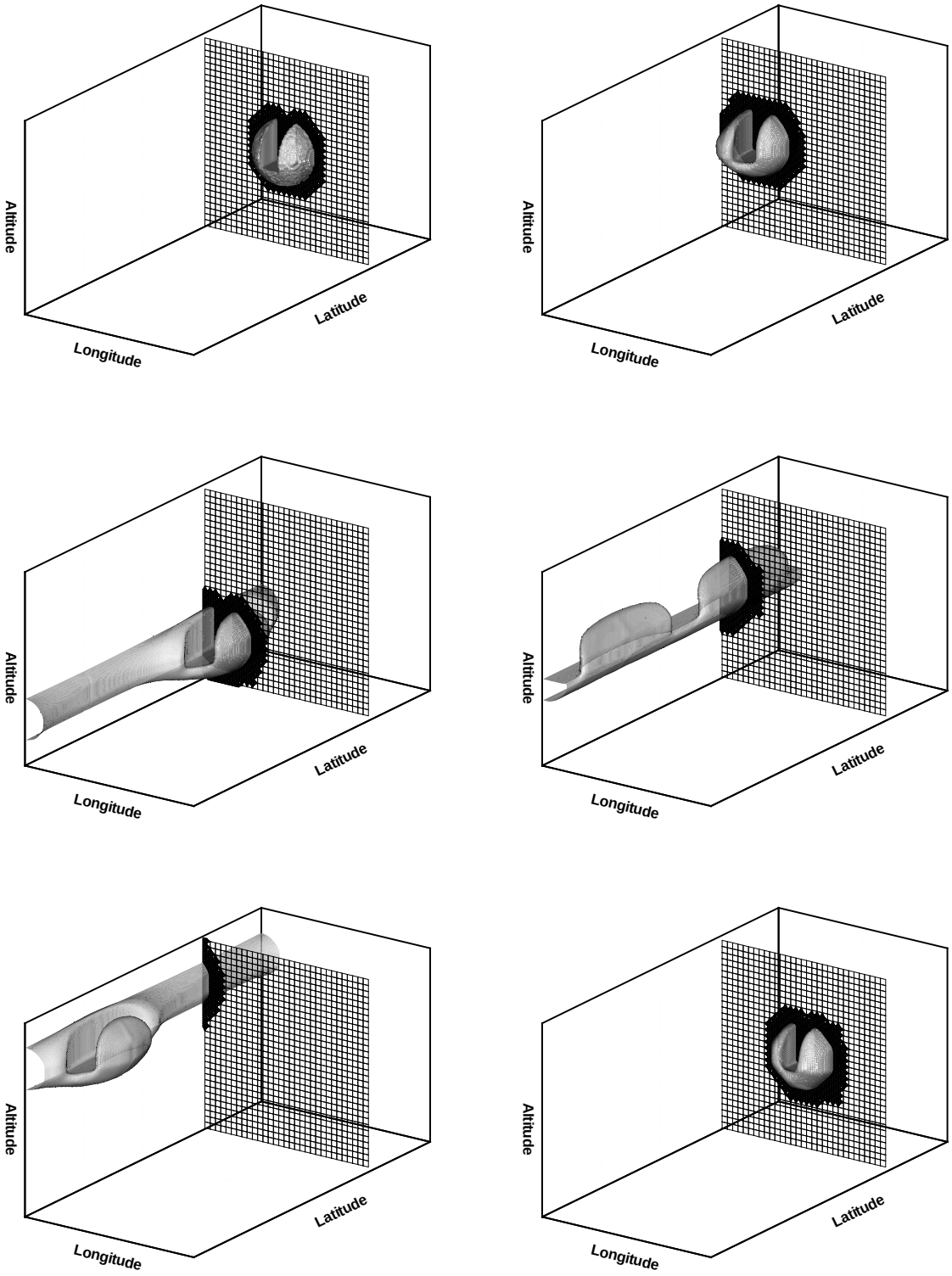

Figure 2: Cylindrical projection of the three-dimensional slotted sphere test case, showing the $\psi = 0.5$ isosurface and the slice of the mesh taken through $\theta = \pi/2$, for the initial conditions (top left), 1/8, 3/16, 1/4, 5/16 of a revolution, and a whole revolution (bottom right).

clearly indicate that AMR can be used to significantly improve the efficiency of flux-based method for approximating the scalar advection equation in practical situations. A speed-up by a factor of 6 is easily obtained in all situations tried so far.

5 Conclusions and Future Work

In this paper an Adaptive Mesh Refinement algorithm has been combined with an appropriately modified, conservative and monotonic finite volume scheme to model three-dimensional tracer advection over the sphere. The resulting method accurately models scalar advection without creating spurious oscillations in the solution, and the addition of Adaptive Mesh Refinement maintains the accuracy of the solution on the finest grid level, while allowing solutions to be obtained with much greater efficiency.

The benefits of using AMR are likely to become even greater with the introduction of additional physical and chemical processes into the model. Initial work has been carried out into the approximation of a shallow water model of the atmosphere and unadapted results obtained using a WAF-type scheme have been encouraging. Work has also begun on the inclusion of chemistry in the model.

6 Acknowledgements

The author would like to thank the UK Universities Global Atmospheric Modelling Project (UGAMP) for funding the work, and Prof. Marsha Berger of the Courant Institute, New York, for supplying the three-dimensional AMR code on which much of the work has been based.

References

[1] J.Bell, M.Berger, J.Saltzman and M.Welcome, Three-dimensional adaptive mesh refinement for hyperbolic conservation laws, *SIAM J. Sci. Comput.*, **15**:127–138, 1994.

[2] M.J.Berger and P.Colella, Local adaptive mesh refinement for shock hydrodynamics, *J. Comput. Phys.*, **82**:67–84, 1989.

[3] M.J.Berger and J.Oliger, Adaptive mesh refinement for hyperbolic partial differential equations, *J. Comput. Phys.*, **53**:482–512, 1984.

[4] R.Nair, J.Côté and A.Staniforth, Cascade interpolation for semi-Lagrangian advection over the sphere, *Q. J. R. Meteorol. Soc.*, **125**:1445–1468, 1999.

[5] E.F.Toro, A weighted average flux method for hyperbolic conservation laws, *Proc. R. Soc. Lond.*, **423**:401–418, 1989.

[6] B.van Leer, Towards the ultimate conservative difference scheme V. A second order sequel to Godunov's method, *J. Comput. Phys.*, **32**:101–136, 1979.

[7] D.L.Williamson, J.B.Drake, J.J.Hack, R.Jakob and P.N.Swartztrauber, A standard test set for numerical approximations to the shallow water equations in spherical geometry, *J. Comput. Phys.*, **102**:211–224, 1992.