
Mesh generation and mesh adaptation for large-scale Earth-system modelling

N. Nikiforakis

Phil. Trans. R. Soc. A 2009 **367**, doi: 10.1098/rsta.2009.0197, published 19 October 2009

References

This article cites 10 articles, 9 of which can be accessed free
<http://rsta.royalsocietypublishing.org/content/367/1907/4473.full.html#ref-list-1>

Article cited in:

<http://rsta.royalsocietypublishing.org/content/367/1907/4473.full.html#related-urls>

Subject collections

Articles on similar topics can be found in the following collections

[climatology](#) (134 articles)

Email alerting service

Receive free email alerts when new articles cite this article - sign up in the box at the top right-hand corner of the article or click [here](#)

INTRODUCTION

Mesh generation and mesh adaptation for large-scale Earth-system modelling

1. Background

The resolution of current operational weather prediction and climate models has significantly increased thanks to the latest generation of supercomputers. However, owing to the hugely disparate length- and time-scales involved, it is still not possible to explicitly resolve many important phenomena, which reduces our ability to numerically predict the short- to medium-term state of the atmosphere and ocean.

Of particular interest are local and often relatively small-scale processes that have a nonlinear effect on larger scales and large-scale but still distinct (localized in space) wave motions that would benefit from increased resolution. Of similar interest are orographic effects (and hence the resolution of topography) and their interaction with weather phenomena.

The obvious solution of a uniform increase of the model resolution is not a viable option because of the associated computational cost in terms of memory as well as time of integration. Grid adaptation methodologies address this problem by locally altering the resolution of the computation in a static or a dynamic way (the latter in response to evolving flow features).

Additional complications arise when grid adaptation has to coexist with a grid generation approach for complex topography (for discretization of orography and bathymetry). Mesh generation is a computational fluid dynamics (CFD) discipline in its own right, because for most engineering applications the shape of the body immersed in (or bounding) the fluid determines the flow around (or inside) it, e.g. flow around an aircraft or inside a nuclear reactor. In the large-scale environmental community, mesh generation research has been largely confined to different approaches to discretizing a spherical surface and more recently to capturing shorelines. Owing to the comparatively low vertical resolution of atmospheric models, discretization of the lower boundary topography has not been a priority and is usually addressed by means of curvilinear (terrain-following) methods. The ever increasing resolution of general circulation models, and hence their ability to better capture significant topography features, has brought the issue of mesh generation to the forefront for a number of reasons, including the fact that poor mesh quality has a detrimental effect on the quality of the flow (e.g. spurious wave generation due

One contribution of 10 to a Theme Issue ‘Mesh generation and mesh adaptation for large-scale Earth-system modelling’.

to highly skewed boundary cells). These problems are well known in the CFD community, and contemporary approaches have recently been employed in oceanic applications, owing to a technology transfer between the communities.

In conclusion, mesh generation and mesh adaptation have extensive use in other disciplines and have recently attracted increased interest in large-scale atmosphere and ocean model development; many researchers are considering approaches to this end, often in partnership with mainstream scientific computing practitioners.

This Theme Issue brings together a number of papers from researchers within the environmental, CFD and scientific computing communities, who are involved in the development and implementation of mesh generation and adaptation methods for atmosphere, ocean and solid Earth applications. Contributions include theory, numerical methods, algorithm development as well as applications. It will be of interest not only to model developers as well as users, but also to those who seek to bridge significantly disparate length scales in their simulations.

2. Space discretization overview and issues

This section summarizes the main approaches for mesh generation and mesh adaptation and raises some of the issues related to space discretization. It is not (and cannot be in such a short space) a full, detailed review of this major area of scientific computing. The reader is encouraged to refer to a number of excellent publications dedicated to this subject (Thompson *et al.* 1999; Plewa *et al.* 2005; Behrens 2006) or textbooks which have large sections on the topic (Tanehill *et al.* 1997; Fergiger & Peric 2002; Hirsch 2007).

This issue is addressed to a multidisciplinary community and with this in mind, I hope that the few paragraphs that follow will give to the non-specialist a taste of the current issues, potential benefits to environmental research and future challenges.

Space discretization is one of the two main components of a numerical simulation (the other one being equation discretization, i.e. the process of approximating the governing partial differential equations by a system of algebraic equations at discrete locations in space and time). It may be naturally subdivided into mesh generation, a process of fitting a computational mesh (grid) around an arbitrary geometry, and mesh adaptation that addresses efficiency issues by concentrating mesh density (and hence resolution) at demanding (e.g. steep gradients) flow areas. Adaptation may further be subdivided into static and dynamic, depending on whether the underlying algorithm has the ability to vary the position of the higher-resolution meshes in time, usually in response to evolving flow features.

(a) Mesh generation

Historically, curvilinear meshes have been employed to discretize arbitrary geometries, as an improvement to step-wise fitting of solid boundaries. Essentially, the technique maps a set of regularly spaced points onto a geometrically complicated domain. However, as the geometry becomes more complex, it is very difficult or impossible to fit a single grid around it. One way around this

problem is to subdivide the domain into geometrically simpler ones, known as blocks. The interfaces of the blocks are matched, and the technique thus is known as *block-structured curvilinear*. If the blocks are allowed to overlap, the resulting meshing approach is known as *chimera*; communication between the blocks is achieved by means of the overlapping regions. Either way, it may be apparent to the reader that these approaches may reach their operational limit as the geometry of the body becomes increasingly complex. Nevertheless, the simplicity of the cell shapes and the underlying data structure means that any other alternative incurs a penalty in complexity, CPU time and storage. Fairly sophisticated packages exist, which are based on this approach (see the article by Henshaw in Thompson *et al.* 1999).

Unstructured grids overcome the problems of structured grids by relaxing the requirement for a logically rectangular cell shape. The elements (or control volumes) may have any shape and there is no restriction on the number of neighbouring elements, which allows for easily generating meshes for arbitrary domains. Although unstructured meshes have a number of disadvantages (irregularity of their data structure, neighbour connections need to be specified explicitly, the matrix of the algebraic equation system does not have a regular diagonal structure, the solvers are slower than those for regular grids), which increase the computational cost compared to their structured counterparts, their flexibility has gained them widespread use in the industrial and academic communities and there are many good packages in operational use.

An interesting hybrid between structured and unstructured meshes is the so-called cut-cell approach (also known as Cartesian-grid, embedded-boundary or shaved-cell methods) that combines the advantages of both the methodologies. In this technique, a regular grid intersects the irregular geometry of the body that is immersed in the flow (e.g. an airfoil) or the edges of the physical domain (e.g. a mountain). While the boundary is accurately represented by a layer of irregular (unstructured) cut cells, regular cells cover the rest of the computational domain away from the solid boundaries.

This approach generates good-quality meshes without the need for a sophisticated package or any additional mesh optimization, has the same low CPU and storage cost as a regular mesh and fast numerical schemes may be used with them. The interesting and critical issues are (i) how to deal with the irregular boundary cells without incurring the penalties of the unstructured approach and (ii) how to avoid the potential severe timestep restriction imposed by the tiny boundary cells that may appear as a result of the cell intersection. Although these issues have been addressed with various degrees of success and there are cut-cell codes currently in use (including an advanced package by NASA; see the article by Aftosmis in Thompson *et al.* 1999), the approach is not mature and it is at the forefront of advanced research in universities and national laboratories.

(b) Mesh adaptation

All of the above approaches to mesh generation have a corresponding mesh adaptation extension. They may be broadly classified according to their time dependency (i.e. whether their position changes in time) as static and dynamic. Static adaptation is mostly suitable for steady (or quasi-steady) flows, whereas dynamic adaptation is the preferred option for distinct evolving flow features.

Another important distinction is whether the approach has the ability to refine in time as well as in space. This is a critical function if the intended use is to bridge massively disparate length scales: the timestep restriction will be prohibitive if the algorithm has no facility to sub-cycle over fine mesh patches that may have a cell size ratio of orders of magnitude to the base coarse mesh cells.

An increase of the mesh density may be achieved by node (or cell) movement or by increasing (and decreasing) the number of cells in time and space. The former is fairly restrictive because one has to know *a priori* the requirements of the flow, so it is better suited for steady-state flows. In fact, one of the first attempts for mesh adaptation was by mesh stretching (effectively a node movement to produce static refinement) to cover the boundary layer of a wing. For most applications, this is a very inefficient way to increase resolution, because the stretching generates very high aspect ratio cells away from the part of the domain that needs to be better resolved.

Research in mesh adaptation has gone a long way since the days of mesh stretching and sophisticated approaches exist for both structured and unstructured cell shapes; a very good representation of the various approaches may, of course, be found in this issue. Good reviews for adaptive meshes can be found in the book by Behrens (2006), albeit mainly for unstructured, triangular meshes, and in the collection of papers by Plewa *et al.* (2005). The latter will also be a good reference for those who are interested in broader scientific computing aspects of the topic.

(c) *Issues and challenges*

The objective of mesh adaptation is to increase computational efficiency without compromising accuracy, i.e. the adaptive simulation at the same effective resolution has to be significantly faster, but as accurate (at least in the domain of interest, which may be a small part of the full computational domain) as the unadapted one. If non-trivial solid boundaries (e.g. lower boundary or shoreline) are part of a simulation, then the mesh adaptation software has to be compatible with standard mesh generation techniques.

Developers of numerical methods for the solution of the governing equations spend a significant amount of time and effort to produce schemes that maintain the mathematical and physical properties of the governing equations of the flow, and the meshing or adaptation approach should not compromise them. As an example, it is highly desirable that a numerical scheme exactly maintains equilibrium (or balanced) solutions. This is important in the case of forcing terms (e.g. terms that express topography in the shallow water equations), where the non-balancing of the terms is known to generate spurious waves, even if the atmosphere (or the ocean) is at rest. Other known potential shortcomings include the generation of spurious errors at the edges of the refinement patches and high overhead of the adaptation-management part of the algorithm. As will become apparent to the reader of the papers within this issue, some approaches have made significant progress towards addressing or eliminating these issues.

In the case of dynamic adaptation, the choice of suitable refinement criteria will determine the success, or otherwise, of the computation (in terms of capturing the features of interest and doing so efficiently). These criteria may be mathematically based (e.g. some absolute or relative error estimate) or heuristic, i.e. determined by the physics and/or chemistry of the flow. The

former tend to incur an additional CPU cost and do not always yield the desired result. Flow-based criteria are the most popular way to flag areas for refinement; these may be composite, i.e. a combination of primitive or conserved variables. The users of the software are likely to have a deep understanding of the physics of the flow under consideration and are thus potentially better suited to determine the optimum refinement criteria than the algorithm developer. It is therefore a task for the algorithm and software developer to design the package in such a way as to allow sufficient flexibility to this end. This is an area where a lot can be learned from other disciplines which have had to address exactly the same issue; fluid dynamics of turbulent combustion is a good example, because there are similarities in terms of the flow features encountered (e.g. convection, large- and small-scale vortical features, jets, etc.). Such a technology transfer was implemented by [Hubbard & Nikiforakis \(2003\)](#), where composite refinement criteria based on vorticity, potential vorticity and location were implemented to automatically fit dynamically adapted fine meshes around the edge of the northern polar vortex and to track the evolution of the tropopause in three dimensions.

Another major issue is the implementation of sub-grid-scale parametrizations; in particular, it is unclear how to handle the interface of variable-resolution regions where a physical process is well resolved in the fine patch, but warrants a sub-grid-scale model in the coarse one. This is another area where a technology transfer from other areas of science and engineering will be of significant benefit, since sub-grid-scale models (e.g. for boundary layers, turbulence, etc.) coexist within adaptive algorithms in many large eddy simulation applications. As in the case of determining refinement criteria, a collaboration between the algorithm developers and the parametrization experts is a good way forward.

There is also a concern on how to perform data assimilation to initialize a model on a multi-resolution mesh based on observed data. This problem is even more important in the case of off-line models that have to be continuously forced with observations, which may not be at the same time intervals as those required by the adaptive model.

The solution to this problem may be different depending on the choice of the adaptive grid methodology, but it is worth referring to the papers by [Hubbard & Nikiforakis \(2003\)](#) and [Nikiforakis \(2005\)](#), which successfully addressed this issue. In those papers, ECMWF re-analysis data sets were used to run a three-dimensional off-line hierarchical adaptive mesh refinement model. The model refines anisotropically in all three dimensions and even though it is not dynamic, the algorithm that initializes the data (and keeps forcing the model) would be the same, irrespective of the equations that are solved. In fact, as the model adapts in time as well as in space (i.e. every fine mesh patch is integrated at its own optimal timestep, all meshes are synchronized at the end of every iteration), the ECMWF data had to be manipulated in a rather involved manner. Nevertheless, the algorithm had a very small CPU-time overhead and the model performed three-dimensional simulations of a stratosphere–troposphere exchange event (initialized on 17 June 1996 and final output on 23 June 1996) at an effective resolution of 12 million cells, over a few hours on a Sun Ultra 10 workstation.

Finally, we remind the reader that an operational piece of software has to adhere to additional considerations, if it is to be of practical use; for example, it has to be amenable to implementation on modern computer architectures.

Depending on the algorithm and on the data structure, some approaches are better suited for parallelization than others. Some of the papers in this issue address this point, which will be critical in the case of operational numerical weather prediction and climate codes.

3. This issue

The editorial objective of this issue has been to cover as broad a range of numerical approaches as possible and also to ensure that the words ‘Earth system’ in the title justified their presence. Thanks to the enthusiastic response from the environmental and the scientific computing communities, it is very pleasing to see that these objectives have been met: there are contributions on a broad range of mesh generation and mesh adaptation techniques, which are implemented on atmosphere, ocean as well as solid Earth applications.

The issue opens with a contribution from Berger *et al.* (in press) on ‘Logically rectangular finite volume methods with adaptive refinement on the sphere’. As the title suggests, the authors consider logically rectangular (i.e. quadrilaterals or hexahedra) hierarchical mesh adaptation on the sphere, using a finite volume (Riemann problem based) approach to solve the governing equations. Their approach employs a logically rectangular grid offering nearly uniform cell-size distribution of computational cells over the spherical surface, which avoids severe time step restrictions. They address the critical issue of exactly maintaining equilibrium solutions, by means of well-balanced methods, which achieve second-order accuracy on smooth solutions, while maintaining the ability to capture discontinuities. Their results compare error norms between the unadapted and the adaptive mesh refinement (AMR) solutions. It is worth noting that there is no spurious error generation at the edges of the refined patches.

The second article by Jablonowski *et al.* (in press) is on a similar block-structured AMR approach on the sphere, employing a finite volume solver: ‘Block-structured adaptive meshes and reduced grids for atmospheric general circulation models’. The discussion is focused on static refinement and reduced grids, which aim to avoid the severe time step restrictions imposed by the use of the latitude–longitude grid in polar regions. This approach is tested on advection, shallow water and three-dimensional hydrostatic problems and the results are quantified by means of error norms against uniform results. They successfully demonstrate speed-up benefits and finally the authors flag mixed-resolution error generation at the fine–coarse interfaces as the subject of future research. It is worth noting the implementation of the algorithm on parallel computer architectures by means of message passage interface (MPI).

The next contribution by Weller (in press) on ‘Predicting mesh density for adaptive modelling of the global atmosphere’ also considers mesh adaptation on a spherical surface, but unlike the two previous articles, it uses polygon-shaped cells for the solution of the shallow water equations. The article focuses on improving the efficiency of the underlying adaptation algorithm by mesh density prediction via solution on a coarse fixed mesh, hence reducing the frequency of re-meshing. As explained by the author, this is an issue of particular importance for unstructured approaches due to the considerable expense of this procedure

and its detrimental effect on accuracy. A number of integrations on a spherical surface demonstrate significant CPU time savings in simulating a barotropically unstable jet.

The development of adaptive mesh approaches inevitably gives rise to problems relating to the varying cell sizes of the computational domain. These problems can only get worse when mesh refinement is combined with a mesh generation approach to discretize the orography (or bathymetry) of the lower boundary. The next two articles employ cut-cell (or Cartesian, embedded boundary) methods for mesh generation and consider gravitationally stratified compressible flows.

In particular, Gatti-Bono and Colella have in the past presented an all-speed projection method that allows the elimination of the CFL restriction on the timestep related to acoustic waves. In their present article, ‘An all-speed projection and filtering method for gravity stratified flows’ (Gatti-Bono & Colella *in press*), they propose a method to isolate those gravity waves that are faster than the underlying fluid advection and employ this filtering to develop an algorithm in which the timestep is constrained only by the advective motions. They demonstrate the effectiveness of their approach by a number of benchmarks, including Lee wave generation in the wake of a ‘witch of Agnesi’ mountain.

Klein *et al.* (*in press*) in their paper on ‘Well-balanced compressible cut-cell simulation of atmospheric flow’ present a novel finite-volume approach for solving the fully compressible equations on cut-cell meshes, which allows for directional operator-splitting and incorporates a well-balanced strategy (cf. our comments for the article by Berger *et al.* (*in press*)). Its effectiveness is demonstrated by means of a number of test cases, including rising thermals and the Lee wave benchmark used previously by Gatti-Bono & Colella (*in press*).

Both methods mentioned above are compatible with hierarchical adaptive mesh refinement, fully account for the arbitrarily shaped boundary cells (a product of using cut cells on arbitrary geometries) and overcome the severe timestep restriction imposed by the tiny boundary cut cells (i.e. their timestep is variable and it is determined by the free-stream cells).

The theme of the next two papers is ocean modelling, using unstructured and triangular meshing, respectively. Piggott *et al.* (*in press*) use fully unstructured meshes to simulate oceanic flows in their paper on ‘Anisotropic mesh adaptivity for multi-scale ocean modelling’. The authors address the issue of the solution of highly anisotropic flows, by employing an optimization-based approach to anisotropic mesh adaptivity. The success of the methodology is demonstrated by means of a number of case studies that compare between results obtained with uniform isotropic resolution, isotropic adaptive resolution and fully anisotropic adaptive resolution.

Behrens & Bader (*in press*) focus on computational efficiency issues in their article on ‘Efficiency considerations in triangular adaptive mesh refinement’, related to the German Indonesian Tsunami Early Warning System. The paper will be of particular interest to the scientific computing part of the community, because it considers the practical and efficient storage and access of the data structures required for triangular adaptive meshing. Quantitative assessment is performed for the solution of Poisson’s equation on simple geometries, whereas meshing for complex geometries is under development.

Tsunami-wave and seismic-wave simulations are considered by Castro *et al.* (in press) in their paper on ‘Space–time adaptive numerical methods for geophysical applications’. Finite volume and discontinuous Galerkin finite element methods are employed on unstructured (triangular) adaptive meshes. An important feature of their approach is the ability to adapt in time as well as in space, i.e. the timestep is not severely restricted by the smallest cell of the computational domain. This advantage is often a feature of hierarchical AMR approaches, but is rare in unstructured ones. The authors also give consideration to the implementation of the algorithm on multi-processor architectures, by subdividing clusters of elements or geological zones with equivalent elements into the desired number of MPI domains. Problems of tsunami- and seismic-wave propagation are used as benchmarks to illustrate the reduction of computational effort required to solve the real-scale problems.

The final article of the issue by Pau *et al.* (in press) on ‘A parallel second-order adaptive mesh algorithm for incompressible flow in porous media’ considers a further solid Earth application with the added complication of multi-phase flow in a porous medium. The paper focuses on the development of a hierarchical, structured adaptive mesh refinement algorithm, which can adapt in time as well as in space (cf. our comments on the article by Castro *et al.* (in press)). The performance of the method and scaling on massively parallel computer architectures (Cray XT4) is quantified by means of multi-component, multi-phase underground flows.

I would like to thank Dr John Bell (Lawrence Berkeley National Laboratory and Managing Editor of Communications in Applied Mathematics and Computational Science) for handling the paper by Klein *et al.* since my name appears in the co-author list. Putting together this issue has been an interesting journey and I would like to thank Suzanne Abbot, at the editorial office of the Royal Society Publications, for her guidance and patience.

N. Nikiforakis

*Department of Physics, Cavendish Laboratory,
University of Cambridge, Cambridge CB3 0HE, UK
E-mail address: nn10005@cam.ac.uk*

References

- Behrens, J. 2006 *Adaptive atmospheric modelling: key techniques in grid generation, data structures and numerical operations with applications*. Springer Lecture Notes in Computational Science and Engineering. Berlin, Germany: Springer.
- Behrens, J. & Bader, M. 2009 Efficiency considerations in triangular adaptive mesh refinement. *Phil. Trans. R. Soc. A* **367**, 4577–4589. (doi:10.1098/rsta.2009.0175)
- Berger, M. J., Calhoun, D. A., Helzel, C. & LeVeque, R. J. 2009 Logically rectangular finite volume methods with adaptive refinement on the sphere. *Phil. Trans. R. Soc. A* **367**, 4483–4496. (doi:10.1098/rsta.2009.0168)
- Castro, C. E., Käser, M. & Toro, E. F. 2009 Space–time adaptive numerical methods for geophysical applications. *Phil. Trans. R. Soc. A* **367**, 4613–4631. (doi:10.1098/rsta.2009.0158)
- Fergiger, J. H. & Peric, M. 2002 *Computational methods for fluid dynamics*. Berlin, Germany: Springer.
- Gatti-Bono, C. & Colella, P. 2009 An all-speed projection and filtering method for gravity-stratified flows. *Phil. Trans. R. Soc. A* **367**, 4543–4558. (doi:10.1098/rsta.2009.0169)

- Hirsch, C. 2007 *Numerical computation of internal and external flows*. Oxford, UK: Butterworth-Heinemann.
- Hubbard, M. E. & Nikiforakis, N. 2003 A three-dimensional adaptive, Godunov-type model for global atmospheric flows. *Mon. Wea. Rev.* **131**, 1848–1864. (doi:10.1175//2568.1)
- Jablonowski, C., Oehmke, R. C. & Stout, Q. F. 2009 Block-structured adaptive meshes and reduced grids for atmospheric general circulation models. *Phil. Trans. R. Soc. A* **367**, 4497–4522. (doi:10.1098/rsta.2009.0150)
- Klein, R., Bates, K. R. & Nikiforakis, N. 2009 Well-balanced compressible cut-cell simulation of atmospheric flow. *Phil. Trans. R. Soc. A* **367**, 4559–4575. (doi:10.1098/rsta.2009.0174)
- Nikiforakis, N. 2005 AMR for global atmospheric modelling. In *Adaptive mesh refinement—theory and applications* (eds T. Plewa, T. Linde & V. W. Weirs). Lecture Notes in Computational Science and Engineering, no. 41. Berlin, Germany: Springer.
- Pau, G. S. H., Almgren, A. S., Bell, J. B. & Lijewski, M. J. 2009 A parallel second-order adaptive mesh algorithm for incompressible flow in porous media. *Phil. Trans. R. Soc. A* **367**, 4633–4654. (doi:10.1098/rsta.2009.0160)
- Piggott, M. D., Farrell, P. E., Wilson, C. R., Gorman, G. J. & Pain, C. C. 2009 Anisotropic mesh adaptivity for multi-scale ocean modelling. *Phil. Trans. R. Soc. A* **367**, 4591–4611. (doi:10.1098/rsta.2009.0155)
- Plewa, T., Linde, T. & Weirs, V. G. 2005 *Adaptive mesh refinement—theory and applications*. Springer Lecture Notes in Computational Science and Engineering. Berlin, Germany: Springer.
- Tanehill, J. C., Anderson, D. A. & Pletcher, R. H. 1997 *Computational fluid mechanics and heat transfer*. Philadelphia, PA: Taylor & Francis.
- Thompson, J. F., Soni, B. K. & Weatherill, N. P. (eds) 1999 *Handbook of grid generation*. Boca Raton, FL: CRC Press.
- Weller, H. 2009 Predicting mesh density for adaptive modelling of the global atmosphere. *Phil. Trans. R. Soc. A* **367**, 4523–4542. (doi:10.1098/rsta.2009.0151)