From the 6th *International Conference on Numerical Grid Generation in Computational Field Simulations*, July 6-9, 1998, Greenwich, UK. International Society of Grid Generation, Mississippi State, MI, 327-335.

A Solution-Adaptive Grid Generation Scheme for Atmospheric Flow Simulations

N. Ahmad T. Dunn D. Mays
D. Bacon M. Hall R. A. Sarma
Z. Boybeyi P. Lee M. Turner.

Science Applications International Corporation
1710 Goodridge Drive, MS T2-3-1
McLean, Virginia 22102
USA
ahmad@apo.saic.com

Abstract

Science Applications International Corporation (SAIC) has developed a novel hazardous dispersion model with support from the Defense Special Weapons Agency. The Operational Multiscale Environment model with Grid Adaptivity (OMEGA) is based on an unstructured prismatic grid. OMEGA has the capability to adapt its grid statically as well as dynamically. The grid can adapt to a set of user defined criteria such as shoreline, terrain, particle locations, and velocity deformation. A considerable reduction in CPU time is achieved by dynamically adapting the grid. This paper gives an overview of the OMEGA model, and describes the grid-generation scheme used by the model. To our knowledge, OMEGA is the only fully operational atmospheric model with spatial and dynamic grid adaptation capabilities.

Introduction

The accurate solution of any complex computational problem depends on fine spatial discretization of the computational domain. The degree of grid refinement is usually constrained by CPU restrictions and by run time expectations. In atmospheric flow simulations, a large amount of CPU time is required by planetary boundary layer physics, cloud microphysics, and radiation transfer calculations. For real-time flow predictions, optimizing the degree of grid refinement, given the CPU constraint, becomes crucial. The use of unstructured grids simplifies this problem by adapting to regions where a high degree of resolution is required, such as shorelines and areas of large terrain gradients, (*e.g.*, see Fig. 1). Further improvement in the solution accuracy can be made by adapting the grid dynamically. Dietachmayer and Droegemeier [1] and Fiedler *et al.* [2] have approached dynamic grid adaptation using structured grids. Their methodologies have been restricted to idealized problems so far and

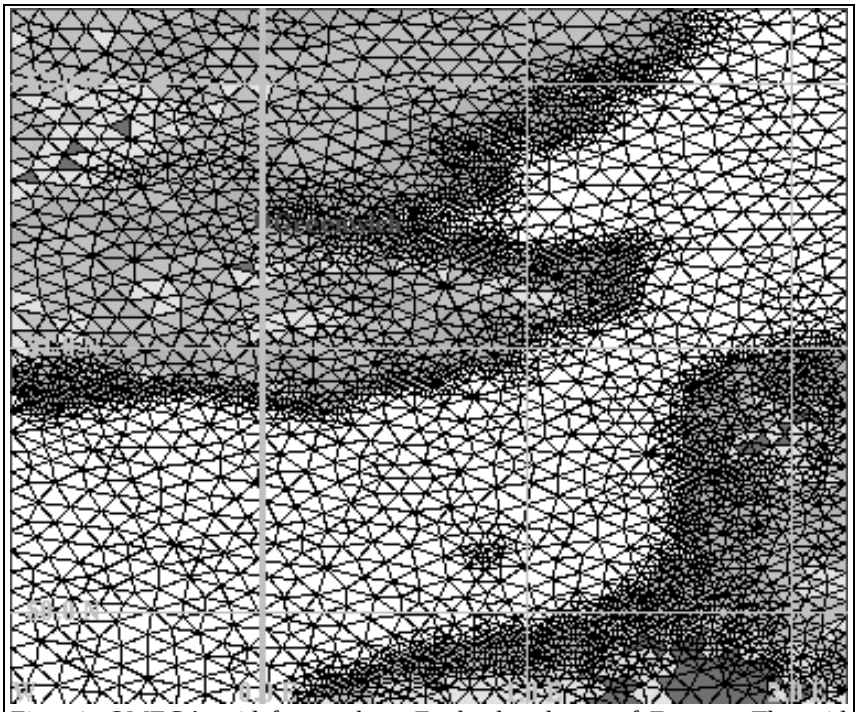

Fig. 1: OMEGA grid for southern England and part of France. The grid refines automatically to resolve the shoreline and steep terrain gradients.

are not suitable for real atmospheric flow simulations which can include complicated terrain features.

Operational Multiscale Environment model with Grid Adaptivity (OMEGA): An Overview

A detailed and comprehensive description of OMEGA is given in [3], [4] and some validation cases are discussed in [5]. For the sake of completeness, however, we describe the OMEGA model briefly here. Apart from the use of unstructured grids, OMEGA is unique because it is fully-compressible and has an embedded atmospheric dispersion model. The discretization of the Navier-Stokes equation is based on Smolarkiewicz's multidimensional positive definite advection transport algorithm [6]. Time-splitting is done by defining areas or masks which require larger or smaller timesteps (depending on cell size). Timestep sub-cycling is performed over smaller cells. Our results have shown a reduction of more than 25 percent in CPU time using this technique while maintaining the CFL criteria. Sub-grid scale diffusion is parameterized using a 2.5 order turbulence kinetic energy (TKE) closure after Mellor and Yamada [7]. Detailed surface physics is included in the model which takes into account the

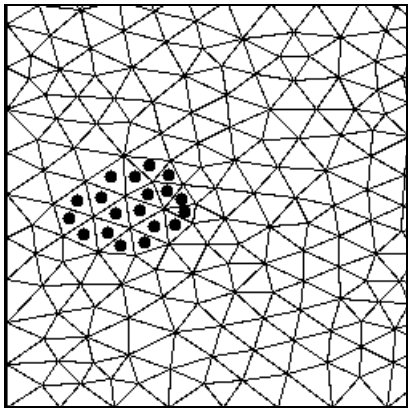

Fig. 2: Cells are tagged for refinement based upon a set of criteria.

inhomogenities of soil-type, vegetation, landuse, sea surface temperature, sub-soil moisture, and sub-soil temperature, *etc*. Both longwave and shortwave radiation are treated in the model to account for heat sinks and sources created by Earth's diurnal cycle. The model also includes cumulus parameterization as well as explicit cloud microphysics. OMEGA's embedded atmospheric dispersion model can be run in Eulerian or Lagrangian mode. An easy-to-use graphical user interface makes pre-processing, grid-generation, model run, and post-processing a user-friendly, seamless and efficient process.

Mesh Generation

OMEGA's mesh generation is based on the algorithm developed by Lottati and Eidelman [8]. User input defines the lateral bounds of the computational domain in terms of latitude and longitude. The grid-generation process starts by creating a rectangular structured grid which is then triangularized. Refinement is based on a set of criteria, which are cast in the form of a weighting function. Cells which are on a land-water boundary (static adaptation) or contain particles (dynamic adaptation) are tagged for refinement (Fig. 2). Tagged cells are then trisected (Fig. 3, left) by adding a vertex in the center of the triangle or bisected by adding a vertex on one of the triangle edges. Refinement by bisecting the edges becomes especially useful for the boundary cells. To improve the quality of new triangles created, the common edge of two adjacent triangles is swapped in a reconnection process (Fig. 3, right). The grid is relaxed by moving the vertices to the center of the polygon surrounding it. Compared to the classic *h*-refinement, this method is faster because it does not require keeping information about the initial grid or grids after each refinement cycle.

The grid is coarsened wherever finer spatial discretization is not essential (*e.g.*, regions of flat terrain or small gradients). Triangles which have bad aspect ratios

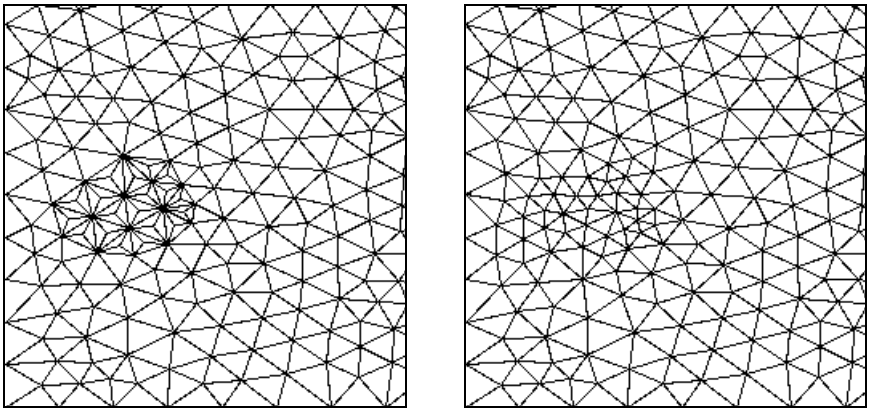

Fig. 3: Tagged cells are first trisected (left), followed by relaxation and reconnection steps (right).

can cause numerical instabilities and also need to be deleted. Coarsening is done by removing a vertex followed by the reconnection and relaxation steps to fill the void.

After the unstructured grid in the horizontal has been generated, the vertices are projected radially upwards from the center of the Earth (see Fig. 4). The vertical grid can have levels of constant or variable heights. For boundary layer flows, it is desirable to have finer resolution in the vertical close to the surface. The stretched grid can be generated by specifying the height of the first level and the stretch ratio. This insures finer vertical resolution near the surface. The vertical grid is terrain following at the surface, but has a constant mean sea level altitude at the top of the computational domain.

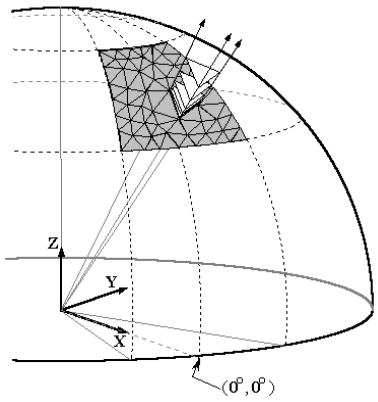

Fig. 4: OMEGA coordinate system.

During the dynamic adaptation, as new cells are generated, all the physical variables are interpolated using a pseudo-Laplacian weighted averaging scheme suggested by Holmes and Connell [9].
The pseudo-Laplacian for a vertex can be defined as,

$$L(\varphi)_o = \sum_{i=1}^{n} w_i(\varphi_i - \varphi_o) \ , \tag{1}$$

stability requires the weights, w_i to be as close to unity as possible,

$$w_i = 1 + \Delta w_i \ , \tag{2}$$

Since, linear functions have zero Laplacian, we can write,

$$L(x)_o = \sum_{i=1}^{n} w_i(x_i - x_o) = 0 \, , \tag{3}$$

and

$$L(y)_o = \sum_{i=1}^{n} w_i(y_i - y_o) = 0 \ , \tag{4}$$

Defining the cost function, C as

$$C = \sum_{i=1}^{n} (\Delta w_i)^2 . \tag{5}$$

We now have an optimization problem of minimizing the cost function given the constraints in equations (3) and (4).

Using Lagrange multipliers we can write,

$$\Delta w_i = \lambda_x(x_i - x_o) + \lambda_y(y_i - y_o) \ , \tag{6}$$

where,

$$\lambda_x = \frac{(I_{xy}R_y - I_{yy}R_x)}{(I_{xx}I_{yy} - I_{xy}^2)} \quad ; \qquad \lambda_y = \frac{(I_{xy}R_x - I_{xx}R_y)}{(I_{xx}I_{yy} - I_{xy}^2)} \ ;$$

and

$$R_x = \sum_{i=1}^{n} (x_i - x_o) \ ; \qquad R_y = \sum_{i=1}^{n} (y_i - y_o) \ ;$$

$$I_{xx} = \sum_{i=1}^{n} (x_i - x_o)^2 \ ; \qquad I_{yy} = \sum_{i=1}^{n} (y_i - y_o)^2 \ ;$$

$$I_{xy} = \sum_{i=1}^{n} (x_i - x_o)(y_i - y_o) .$$

Physical variables which are cell-centered are interpolated to vertices before each adaptation cycle using the pseudo-Laplacian weighted averaging scheme described above. As new cells are created or removed locally a simple linear interpolation is done to assign values to new vertices and cell centers. All the

diagnostic variables are calculated for new cells, particle locations array is updated, and the mesh is re-masked to calculate new time steps for the time-splitting scheme. At the beginning of a simulation the minimum elevation is saved. As the grid is refined or coarsened, no cell is allowed to have elevation lower than the initial minimum elevation value. This is to insure that no extrapolation is done. The original base state variables are also saved and as new cells are created, base state profiles are generated for these cells using the initial base state.

Simulating Atmospheric Dispersion of Particles

Two simulations were run to demonstrate the advantage of dynamic grid adaptation. The simulation domain extended from -98.65 degrees to -95.65 degrees in the west-east direction and from 41.0 degrees to 45.0 degrees in the north-south direction, covering parts of Nebraska, Iowa, Minnesota and South Dakota. The terrain for this test simulation was relatively flat. Terrain data with a resolution of 30 seconds and shoreline data with a resolution of 5 minute was used. In the vertical a stretched grid consisting of 30 levels was generated. The first level near earth's surface had a thickness of 15 meters and further levels were created using a stretch ratio of 1.15. A particle source was defined in the south-west corner of the domain with an emission rate of 5 particles per minute. The simulations were initialized using U.S. Navy's NOGAPS gridded data. The simulations were run out to 12 hours.

In the first simulation (Case A), a variable grid resolution of 5 km to 15 km was specified (see Fig. 5, left). The initial grid was not modified during the simulation.

In the second simulation (Case B) a variable resolution of 10 km to 36 km was prescribed for the initial grid generated (see Fig. 5, right). The adaptation criteria was set to the particle location. An adaptation cycle was invoked after every one hour of simulation time. Cells were tagged by defining a Gaussian function around the particles. Tagged cells were refined and all physical fields were interpolated at each adaptation cycle.

Fig. 6 (left) shows the plume at 9 hours into the simulation for the fine grid. The flow is mostly south-westerly. Fig. 6 (right) shows the plume for the adapted grid simulation at the same time. Around nine hours into the simulation the winds shift slightly towards east (see Fig. 7). As the plume shifts the adaptation process follows. Fig. 7 (right) shows the adapted grid at 12 hours into the simulation. The final adapted grid has 1266 cells and the same flow field. The resolution of the final adapted grid varied from 7 km to 38 km. A reduction of over 40 percent in CPU time was achieved in this simulation.

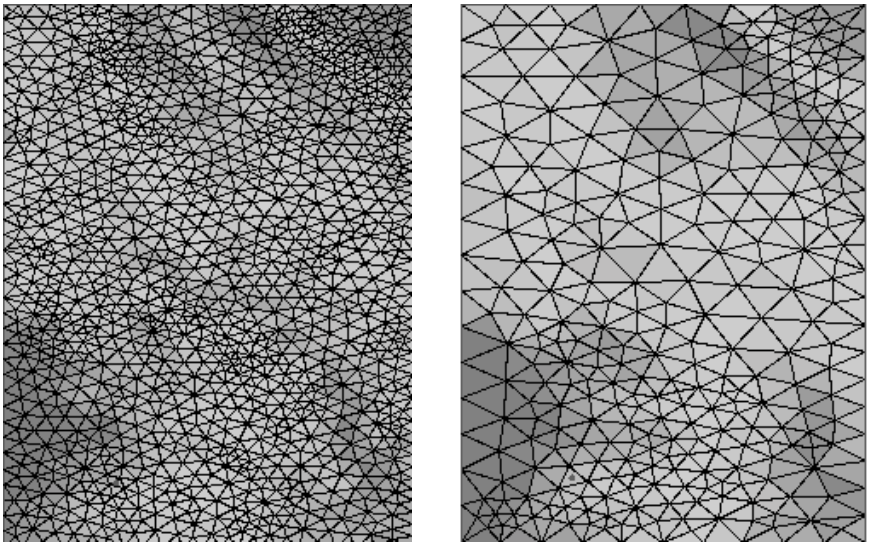

Fig. 5: (left) Case A simulation grid (2335 cells). (right) Case B Simulation initial grid (582 cells). Darker areas represent higher elevation and lighter areas represent low elevation.

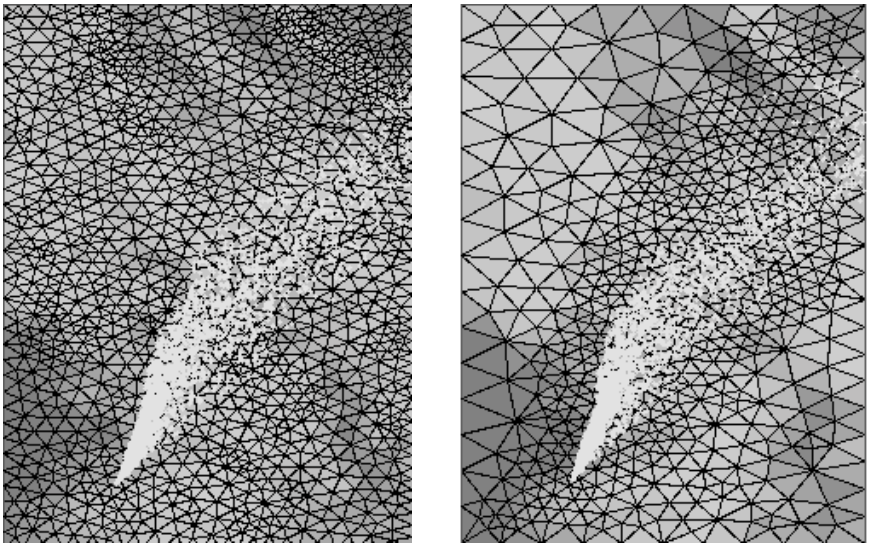

Fig. 6: Case A simulation (left) and Case B simulation (right) showing the particles after 9 hours. The grid in Case B has adapted in regions of particles.

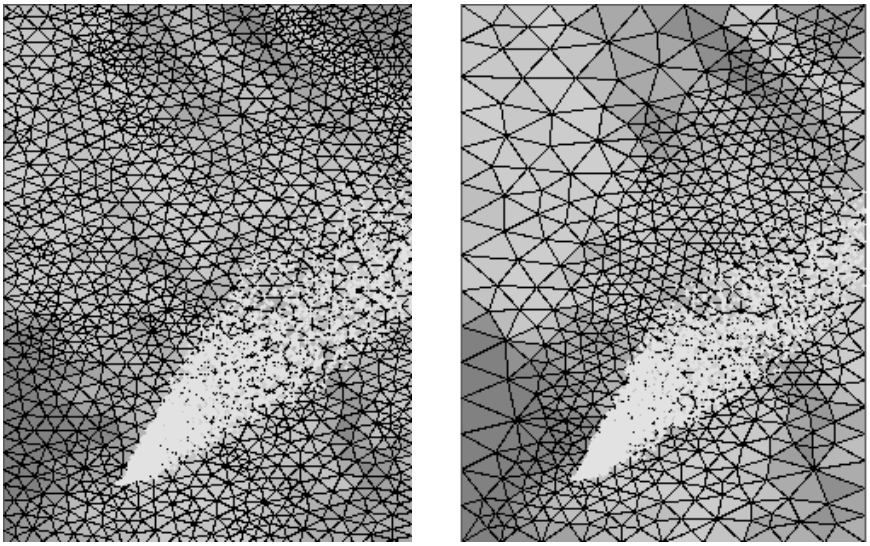

Fig. 7: Case A simulation (left) and Case B simulation (right) showing the particles after 12 hours. The final grid in Case B has 1266 cells.

During dynamic adaptation, the coarsening cycle was turned off and cells were not removed from low activity regions. Further reduction in CPU time can be achieved by coarsening the cells in regions of low activity.

Conclusions

Unstructured adaptive grids are used for simulating complicated atmospheric processes in an operational setting. The mesh-generation algorithm is efficient and well-suited for atmospheric flow simulations. Reduction in CPU time (over 40 percent) is achieved by dynamically adapting the grid during the simulation. We have shown a simulation where the grid adapts to particles and reproduces the same field as in the case of global refinement. The algorithm is robust and a set of other criteria can also be defined for refining/coarsening the cells, e.g., velocity deformation and potential temperature gradient. The technique shows promise in accurately simulating in real-time, emergency response scenarios, such as hurricane or release of hazardous materials.

Acknowledgment

This work has been supported by the US Defense Special Weapons Agency under contracts DNA001-92-C-0076 and DNA001-92-C-0130.

References

[1] Dietachmayer, G. S. and K. K. Droegemeier, 1992: Application of Continuos Dynamic Grid Adaption Techniques to Meteorological Modeling. Part I: Basic Formulation and Accuracy. *Mon. Wea. Rew.*, **120**, 1675-1706.

[2] Fiedler, B., Z. Huo and P. Zhang, 1998: Dynamic Grid Adaption for Atmospheric Boundary Layers. *Proc. of the 12th Conf. on Num. Wea. Prediction*. AMS. 253-254.

[3] Bacon, D. P., *et al*., 1995: The Operational Multiscale Environment Model with Grid Adaptivity. Defense Nuclear Agency. DNA-TR-95-30.

[4] Bacon, D. P., *et al*., 1996: An Overview of the Operational Multiscale Environment model with Grid Adaptivity. *Proc. 15th Conf. on Wea. Anal. and For*., AMS.

[5] Boybeyi, Z., *et al*., 1997: Recent Validation of the Operational Multiscale Environment Model with Grid Adaptivity. *12th Symposium on Boundary Layers and Turbulence*. AMS. 227-228.

[6] Smolarkiewicz, P. K., 1984: A fully multidimensional positive definite advection transport algorithm with small implicit diffusion. *J. Comput. Phys.*, **54**, 325.

[7] Mellor, G. L. and T. Yamada, 1974: A hierarchy of turbulence closure models for planetary boundary layers. *J. Atmos. Sci*., **31**, 1791-1806.

[8] Lottati, I. and S. Eidelman, 1994: A second-order Godunov scheme on a spatial adapted triangular grid. *Applied Numerical Mathematics*, **14**, 353-365.

[9] Holmes, D. G. and S. D. Connell, 1989: Solution of the 2D Navier-Stokes Equations on Unstructured Adaptive Grids. AIAA. Paper AIAA-89-1932.