

Delaunay Mesh Generation for an Unstructured-Grid Ocean General Circulation Model

S. Legrand^{1,2}, V. Legat² and E. Deleersnijder¹

¹ Institut d'Astronomie et de Géophysique G. Lemaître, Université Catholique de Louvain,
2 Chemin du Cyclotron, B-1348 Louvain-la-Neuve, Belgium

² Centre for Systems Engineering and Applied Mechanics, Université Catholique de Louvain,
4 Avenue Georges Lemaître, B-1348 Louvain-la-Neuve, Belgium

E-mail: legrand@astr.ucl.ac.be

Abstract

An incremental method is presented to generate automatically boundary-fitted Delaunay triangulations of the global ocean. The method takes into account Earth curvature and allows local mesh refinement in order to resolve topological or dynamical features like midocean ridges or western boundary currents. Crucial issues like the nodes insertion process, the boundary integrity problem or the creation of inner nodes are explained. Finally, the quality of generated triangulations is discussed.

Key words: unstructured grid, Delaunay triangulation, sphere, global ocean.

Accepted for publication in Ocean Modelling, June 2000.

1 Introduction

Since the pioneering work of Bryan (1969), most ocean general circulation models (OGCMs) are using finite-difference techniques on structured grids. If those grids are based on the geographical coordinates, stability problems are likely to arise in the vicinity of the North Pole (Williamson, 1979). Many approaches were suggested to deal with this difficulty within the framework of structured grid models (e.g. Williamson, 1979, Murray, 1996). So far, essentially three types of solution have been implemented in OGCMs: combined grids as equatorial transform (Deleersnijder et al., 1993, Eby and Holloway, 1994, Coward et al., 1994), grids generated semi-analytically (Madec and Imbard, 1996) and grids generated analytically (Murray, 1996, Purser and Rančić, 1997, Bentsen et al., 1999).

However, two main drawbacks due to the rigidity of all these structured grids are inescapable. Firstly, their staircase representation of coastlines exerts some spurious form stress on model boundary currents (Adcroft and Marshall, 1998) and the alternative grid generation method based on boundary-fitted coordinates (Wilkin and Hedström, 1998) only works for regional applications. Secondly, the rigidity of these grids combined with the expensive CPU cost of the OGCMs prevents the resolution of relatively small topological and dynamical features without nesting or adaptive mesh refinement (Blayo and Debreu, 1999). However, the fine resolution of these features is likely to be a key point for a globally well-resolved ocean circulation model. Among these are equatorial dynamics, western boundary currents, mesoscale eddies, ridges, continental slopes, channels and straits.

Using unstructured grids would help to solve the problems listed above. This is why numerical methods as finite elements may be potentially interesting for modelling the global ocean. While these were successfully used in tidal (Shum et al., 1997) and coastal modelling (Lynch et al., 1996), little interest has been expressed so far in the other branches of oceanography (Myers and Weaver, 1995, Le Roux et al., 1998) apart from the spectral finite element ocean model (SEOM) (Iskandarani and Haidvogel, 1995, Curchister and al., 1998). However, they have significant advantages over finite difference.

A first advantage of finite element methods is their inherent ability to deal with unstructured grids. These grids easily allow local grid refinement to give high resolution without loss of accuracy. The use of such grids also avoids the meridian convergence problem of longitude-latitude grid in the vicinity of the North Pole.

A second advantage is their strong and rigorous mathematical foundation based on a weighted residual formulation. The solution which is typically constructed from a polynomial expansion belongs to an a priori defined function space. The mathematical foundations of the finite element method also allows a precise definition of the accuracy which leads to adaptive finite element methods.

In this paper, we present a method to generate automatically unstructured meshes of the global ocean. To take advantage of robust and well-known algorithms, we have subdivided the ocean into a conform triangulation. Such triangulations have been already used in oceanography and are generated by usual codes like TriGrid (Henry and Walters, 1993). These codes – which are principally aimed toward the automatic generation of finite element networks in the two horizontal dimensions for use with models of coastal circulation – are not adapted to the global ocean, a domain for which Earth curvature influences dramatically the quality of generated meshes. For instance, to reach good quality meshes of the global ocean with TriGrid, Le Provost et al. (1994) had to split the global ocean into subdomains which are supposed plane. Therefore, we developed our generator to provide a mesh generator well adapted to large-scale oceanography. From a technical point of view, the ocean surface is directly triangulated on spherical surface. We have also implemented the automatic creation of nodes – both boundary and inner nodes – during the mesh generation.

A first Delaunay triangulation generator on the sphere using an incremental method was implemented by Renka (1997) but the applications of this generator did not deal with boundaries like the coastlines. This issue is properly addressed below.

2 Unstructured adaptive finite element methods

Let \mathbf{u} be the exact solution of a general circulation model. The block-field vector \mathbf{u} includes all variables of an oceanic circulation model : the velocity components, the pressure, the temperature and the salinity. Let \mathbf{u}^h be the approximation of \mathbf{u} so that

$$\mathbf{u}^h(\mathbf{x}, t) = \sum_{i=1}^N \mathbf{U}_i(t) \phi_i(\mathbf{x}) \quad (1)$$

where \mathbf{U}_i are unknown nodal values and where ϕ_i are given piece-wise polynomial shape functions. This form of \mathbf{u}^h allows a finite element treatment of the spatial discretization and a finite-difference treatment of the time discretization.

For a coercive discrete operator (typically, a coercive operator can be viewed as the discrete algebraic system obtained by applying standard Galerkin finite element procedure to a dissipative physical model), the following interpolation property holds (Ciarlet, 1979) :

$$\|\mathbf{u} - \mathbf{u}^h\|_{H^1(\Omega)} \leq C \frac{h^{p+1}}{\rho} \|\mathbf{u}\|_{H^2(\Omega)} \quad (2)$$

where $\|\cdot\|_{H^1(\Omega)}$ and $\|\cdot\|_{H^2(\Omega)}$ are usual Sobolev norms, p is the order of the shape functions, h is a typical mesh size, ρ is a typical inscribed circle radius of the triangulation elements, C is a triangulation-independent constant. For a given mesh size, the fact that the triangulation exhibits small shape factors $\frac{h}{\rho}$ is a crucial issue for the accuracy of the finite element approximation so that the normalized inverse of the shape factor is used to quantify the quality of a triangulation. This quality factor belongs to the interval $[0, 1]$. The quality factor of a degenerated triangle vanishes while that of an equilateral triangle is equal to unity. It is commonly accepted that a triangulation is a good one if all its triangles have just acute angles. It means that the quality factor of the worst triangle must be greater than 0.5.

Equation (2) is an a priori error estimation for all problems where a dissipative term appears, such as shallow water models (see, for instance, Foreman, 1984). A more accurate a posteriori error estimation can be calculated when the finite element solution \mathbf{u}^h is known. The joint use of a priori and a posteriori error estimations leads to an automatic adaptive method (Johnson and Szepessy, 1995). In other words, an adaptive method consists in calculating a discrete solution \mathbf{u}^h and the a posteriori error on a first grid. If the error is greater than the precision objective, the a priori error is used to determine the minimum number of nodes which must be added in the grid to reach the objective. Finally, a discrete solution is calculated on the new grid and the process starts again. Here, we only investigate the initial step, i.e. the design of a first mesh. To obtain such a mesh with suitable shape factor, we use the standard Delaunay triangulation.

3 Delaunay triangulation on the sphere

Let $X = \{x_1, x_2, \dots, x_n\}$ be a set of points in a plane which are called nodes. Now, partition the plane by assigning every point in the plane to its nearest node. All those

points assigned to x_i form the Voronoi region $V(x_i)$. The set of all points belonging to more than one Voronoi region defines the Voronoi diagram. The dual graph of a Voronoi diagram is obtained by drawing connecting lines between nodes perpendicular to the edges of the diagram (Figure 1). It can be proved that the dual graph of a Voronoi diagram produces a triangulation of the nodes which is called the Delaunay triangulation. This duality between Delaunay triangulation and Voronoi diagram ensures the uniqueness of the Delaunay triangulation – if co-cyclic points are excepted.

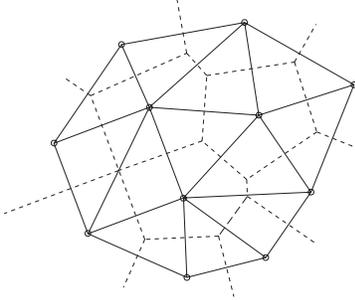


Figure 1: Delaunay triangulations (solid lines) are obtained by drawing connecting lines between nodes perpendicular to the edges of the Voronoi diagram (dashed lines). When four nodes are co-cyclic, quadrangular elements are produced but they can be correctly divided into two triangles.

An important property holds for the Delaunay triangulation : the open circle circumscribed to a triangle does not contain any triangle vertex. One of the consequences of this property is that Delaunay triangles exhibit good shape factors.

To adapt Delaunay triangulation to the sphere, the usual distance between two points may be replaced by the geodesic one, i.e. the length of the unique great circle arc passing through two points of the sphere. Therefore, we are able to define the Voronoi diagram on the sphere. It can be shown that the dual triangulation of this diagram has in most cases good shape factors. Without loss of generality, such a triangulation can be characterized by the Delaunay criterion: if S is the surface to triangulate, the open sphere circumscribed to any triangle of the triangulation and whose center lies on the surface of S does not contain any triangle vertex (Jayaraman et al., 1997).

Delaunay triangulations are used for several applications and a variety of algorithms to obtain it was developed. In the intersection of halfplanes approach, each Voronoi region is constructed separately, by intersecting $n - 1$ halfplanes. This algorithm has a cost of $O(n^2 \log n)$. The best algorithm from a theoretical point of view is the divide and conquer approach proposed by Bentley and Shamos (1976). This approach has a complexity of $O(n \log n)$ but it is rather difficult to implement. An equivalent approach but more simple to implement is the clever plane-sweep algorithm of Fortune (1987). This algorithm passes a sweep line over the plane, leaving at any time the problem solved for the portion of the plane already swept, and unsolved for the portion not yet reached. Actually, this very short description hides several major problems, which Fortune surmounted by an extraordinary clever idea (O'Rourke, 1993). A popular approach is the incremental construction of the Delaunay triangulation. Suppose a Delaunay triangulation \mathcal{T} of k

nodes is already constructed, and now we would like to construct the triangulation \mathcal{T}' , after adding the node p . Suppose p falls inside the circles associated with several triangles in \mathcal{T} . Then, these triangles cannot be triangles in \mathcal{T}' anymore, because they violate the Delaunay criterion. It turns out that these are the only triangles of \mathcal{T} that are not carried over to \mathcal{T}' and that these triangles are located on one area of the graph. The algorithm complexity is $O(n^2)$ and is known as the incremental Watson algorithm (Watson, 1981).

As our problem consists in creating a good triangulation from a given spherical domain but without knowing a priori all the nodes x_i , we have adapted the incremental algorithm with automatic creation of nodes on segments in a dynamic fashion in order to reach a preset level of local size for the element (Figure 2):

- *Creation of an initial triangulation.* As for all incremental methods, the Watson algorithm needs a starting point, i.e. an initial Delaunay triangulation which can be easily implemented. For symmetry reasons we have chosen an initial triangulation with 5 nodes (two on the poles and the remaining three on the equator) and six triangles. However, this is not the only possible choice.
- *Insertion of boundary nodes.* To have an easy representation of continents and islands, known boundary nodes are firstly inserted in the triangulation one by one. As the Delaunay triangulation is unique and if numerical errors are ignored, the order of nodes insertion does not influence the final Delaunay triangulation.

The creation of a Delaunay triangulation \mathcal{T}_{k+1} of $k+1$ nodes from a Delaunay triangulation \mathcal{T}_k of k nodes requires two steps. The first one consists in inserting the new node p in the old triangulation \mathcal{T}_k : the triangle which contains p is searched and replaced by three new triangles whose vertices are the new node p and the vertices of the old triangle. The second step transforms the new triangulation into a Delaunay one. Only a limited number of triangles are involved by such a transformation. This is based on the two following properties. The first one says that if the Delaunay criterion is respected for all configuration of two adjacent triangles so the triangulation is a Delaunay one while the second one says that a triangulation of a set of nodes X can be transformed into another triangulation of X by a succession of segment swaps. Each swap creates a new triangulation. A Delaunay triangulation can be obtained only by swapping common segments of triangles which do not respect the Delaunay triangulation (Figure 3). More details can be found in Cherfils and Hemerline (1990).

Errors due to floating point computations may lead to defects such as intersecting and overlapping triangles. Typically, these defects occur when the test for finding the triangle containing the node to insert fails (Boender, 1994). Therefore to identify if a node is at the right or the left of a segment, the implementation must be done in such a way that the result is robust: all nodes must belong only to one triangle in a conform triangulation.

- *Boundary integrity problem.* To represent boundaries, the Delaunay triangulation is constrained to avoid the swap of a boundary segment. Once created, they will never be destroyed. This constraint can already be applied during the insertion of

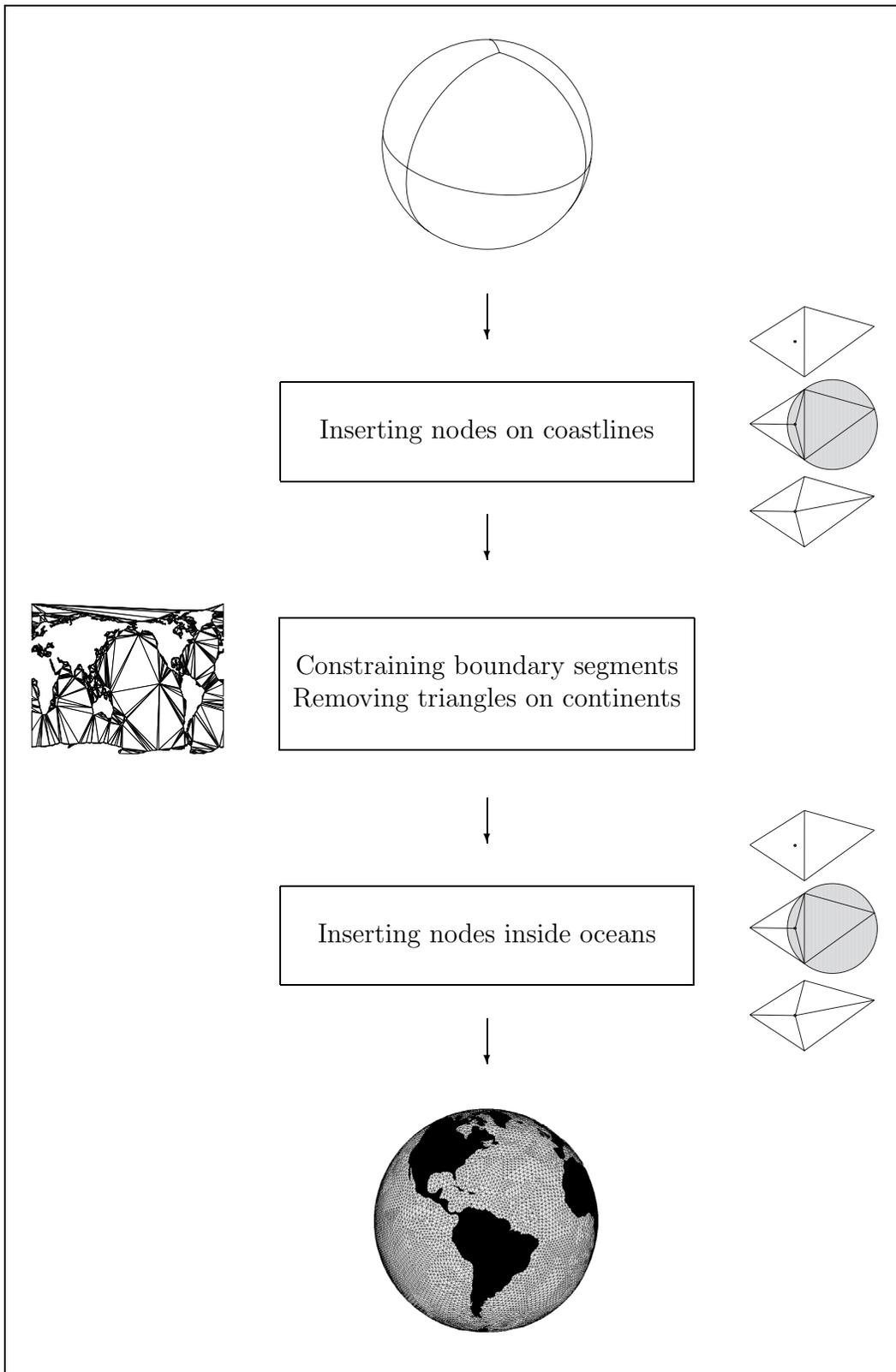


Figure 2: Basic steps of the Delaunay triangulation algorithm.

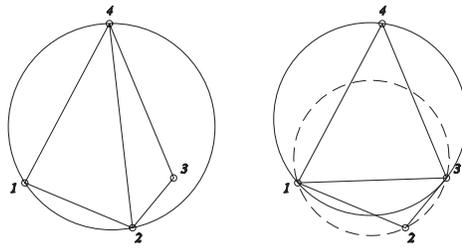


Figure 3: Swapping a segment to obtain triangles satisfying the Delaunay criterion.

boundary nodes. However, when all boundary nodes are inserted in the triangulation, nothing ensures that all the boundary segments have been created. This is the boundary integrity problem. The easiest solution to implement is certainly to find the missing segments and to subdivide them by inserting their midpoints so, after some repetitions, the boundaries are closed. By this approach, very short segments can be created so that at the end of the algorithm bad quality triangles could be generated. Another solution without insertion of new nodes consists in swapping the edges intersecting the missing boundary segments until it is created. Then, it must be verified that the new triangles respect the Delaunay criterion.

It could also be useful to extend this algorithm so that segments are created taking into account information about bathymetry.

- *Destruction of triangles.* The insertion of the boundary nodes in the triangulation generates triangles outside the ocean. To avoid the creation of undesired inner nodes, those are destroyed immediately after the insertion of the boundary nodes (Figure 4).
- *Creation of inner nodes.* Finally, new nodes are created in a dynamic fashion at the middle of well chosen segments in order to reach a preset level of local mesh size. Uniform distribution of triangle shapes is obtained by an iterative procedure that consists in inserting a new node in the middle of the longest segment of the triangulation. Non-uniform triangulations are obtained by using a weighted distance to identify the longest segment. Of course, after each node insertion, the new triangulation is transformed into a Delaunay one. These algorithms have the main advantage that they bypass problems due to floating point computations. Illustrations of uniform and graded meshes along oceanic western boundaries and Equator are displayed in Figure 5.

4 Results

Typically, the CPU cost of the triangulation algorithm is totally negligible with respect to that of OGCM calculations. However, we provide here some CPU times for completeness. On a standard PC (Pentium III - 500 MHz - 120 Mo of RAM), the time needed to generate an unstructured graded mesh of 10^5 nodes is 12 seconds. It varies linearly with the number

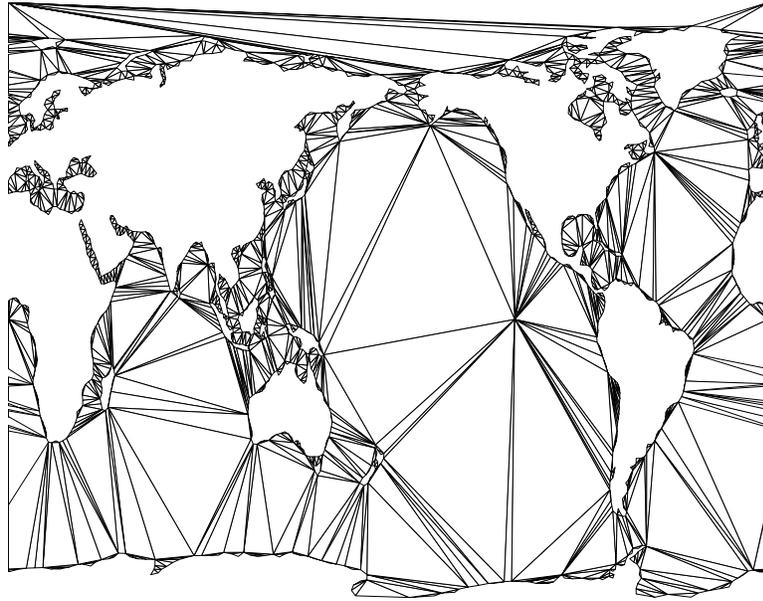


Figure 4: Triangulation after that triangles on continents were removed: the coastlines are well-fitted, particularly the European and the Indonesian coastlines. Obviously, the shape of most triangles is very far from the equilateral.

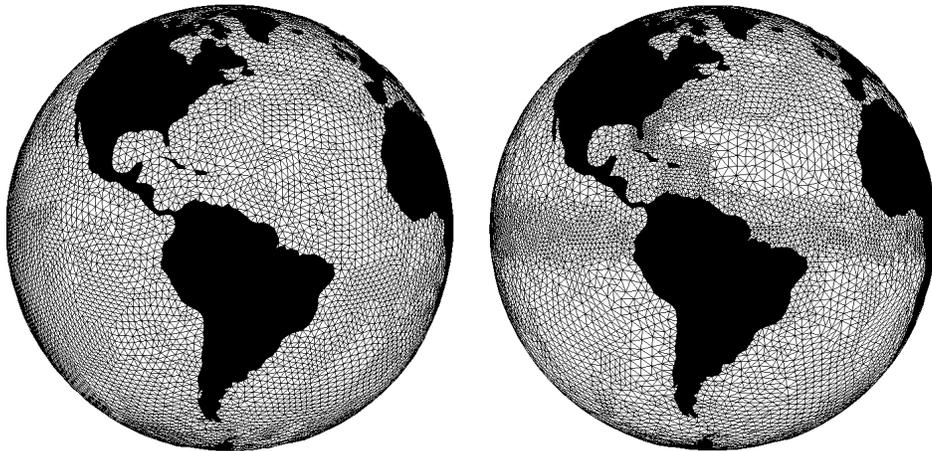


Figure 5: Both triangulations contain approximately 10000 nodes. The first triangulation has an almost uniform mesh size of 130 km. The second triangulation is adapted to resolve the Gulf Stream dynamics. Its typical mesh size is equal to 20 km in the refined regions and is equal to 160 km elsewhere.

of inserted nodes. Our adapted implementation allows to obtain such a result which is quite better than the theoretical complexity of $O(n^2)$. We use a binary tree which allows at any time the knowledge of the longest weighted segment: this bypasses most of the searches. Of course, such a data structure increases memory requirements. For example, the insertion of a new node requires 560 bytes.

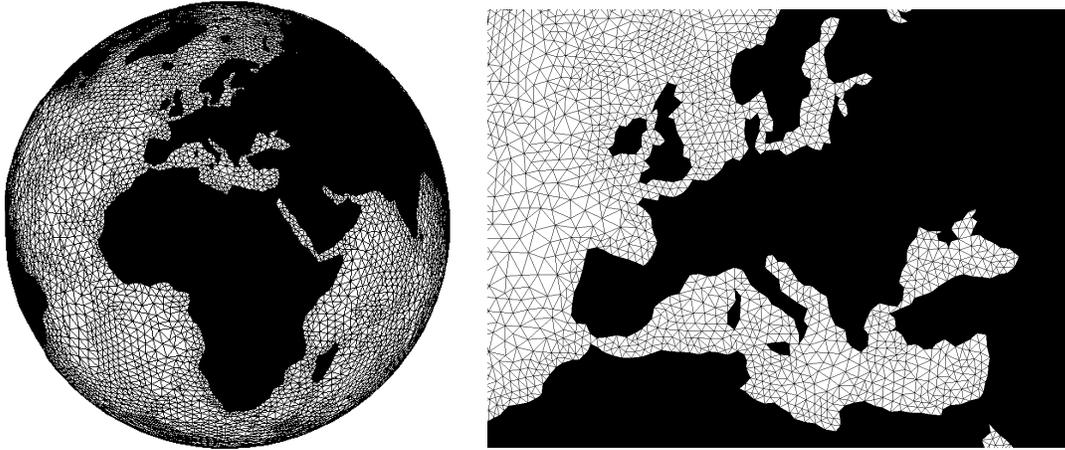


Figure 6: Triangulation whose refinement is based on the bathymetry and close up view of Europe area.

The second issue is the quality of the triangulations which is given by the normalized inverse of the shape factor. The boundary design and the segments weight govern essentially the quality of the grid — both controlled by the user.

The boundary design must be adapted to the desired mesh size: a too fine boundary design compared with the typical mesh size leads to flat triangles. The stopping condition of the incremental algorithm must be based on an a priori specified mesh size which allows an automatic fit of the boundary design. To obtain a good representation of the coastlines, we treat the boundary design in two steps. In the first step, a physical representation of the coastlines is obtained by inserting boundary nodes with exact geographical coordinates. The choice of these nodes has to be driven both by the geometry of the boundaries and by modelling issues. In the second step, such a representation can be completed by adding new boundary nodes with interpolated coordinates and not exact geographical coordinates. In fact, this step filters small-scale details of the coastlines which have been previously judged irrelevant for the calculations and which would introduce critical numerical difficulties (bad triangles or introduction of significant errors due to the coarse representation of the features). In Figure 6, the linear piecewise interpolation of the coastlines could be improved by using high order piecewise polynomial interpolation (typically cubic splines or Bezier curves).

Strong differences in weight distribution which lead to highly graded meshes influence poorly the quality of the generated meshes. Indeed, sharp gradients in weight distribution mean that two regions with different mesh size are juxtaposed so that at the interface flat triangles could be generated. For instance, an abrupt step trough which the segments weight is doubled creates triangles whose quality factor is smaller than 0.4.

Figure 7 shows the distribution of the quality factor of the mesh produced by the Watson algorithm with automatic creation of nodes on segments. More than 90 % of the triangles of the grid have quality factor greater than 0.7 and any triangles have quality factor smaller than 0.5. The worst quality factor is equal to 0.53. Although this distribution is already satisfactory (cf. section 2), it can be improved by using the Laplacian smoothing and the mesh relaxation (Boender, 1994). In fact, the first method optimizes nodes position while the second optimizes the connexion pattern between the nodes. Clearly, many triangles in the modified mesh exhibit better shape factor. In other words, these are closer to equilateral triangles. However, a very few number of elements (0.5%) have still a relatively poor quality factor in particular near the coastlines.

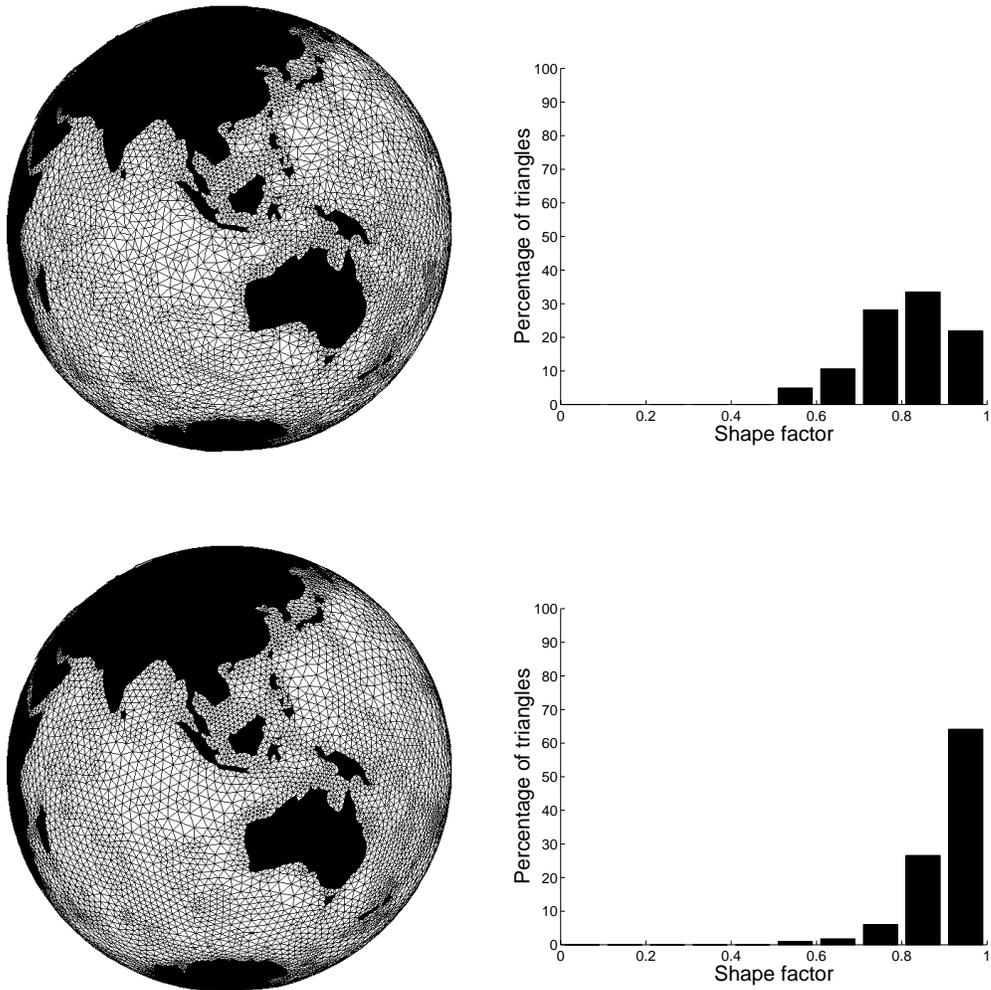


Figure 7: Improvement of shape factor distribution by Laplacian smoothing and mesh relaxation.

5 Conclusion

We have implemented an incremental algorithm to generate unstructured meshes for a global ocean circulation model. Needing only the specification of coastlines and segments weight, a triangulation with good shape factors can automatically be created. In particular, no singularities or uncontrolled convergence zones are created. The generator appears to be able to refine at a correct scale the topological and dynamical features which are key points for a globally well-resolved ocean circulation model. Among these are equatorial dynamic, western boundary currents, mesoscale eddies, ridges, continental slopes, channels or straits.

The next step of our work will be the development of an ocean general circulation model based on the finite element method to compare the efficiency of unstructured grids with classical approaches.

Finally, the mesh generator is a general purpose tool that could be useful in other fields of geophysics. Let us just mention the interpretation of scattered measurements on the Earth (Nielson, 1993).

Acknowledgements

We are indebted to J.-M. Campin, H. Goosse and O. Magotte for useful discussions. Eric Deleersnijder is a Research Associate with the National Fund for Scientific Research of Belgium. The present work is funded by the Convention d'Actions de Recherche Concertées ARC 97/02-208 (Communauté française de Belgique).

References

- Adcroft, A. and D. Marshall (1998) How slippery are piecewise-constant coastlines in numerical models? *Tellus*, 50 A, pp. 95–108.
- Bentley, J. and M. Shamos (1976) Divide-and-conquer in multidimensional space. Conference Record of the Eighth Annual ACM Symposium on Theory of Computing, pages 220–230.
- Bentsen, M., G. Evensen, H. Drange and A.D. Jenkins (1999) Coordinate transformation on a sphere using conformal mapping. *Monthly Weather Review*, 127(12), pp. 2733–2740.
- Blayo, E. and L. Debreu (1999) Adaptive mesh refinement for finite-difference ocean models: First experiments. *Journal of Physical Oceanography*, 29(6), pp. 1239–1250.
- Boender, E. (1994) Reliable delaunay-based mesh generation and mesh improvement. *Communications in Numerical Methods in Engineering*, 10, pp. 773–783.

- Bryan, K. (1969) A numerical Method for the study of the circulation of the World Ocean. *Journal of Computational Physics*, 4, 347-376.
- Cherfils, C. and F. Hermeline (1990) Diagonal swap procedures and characterizations of 2d delaunay triangulations. *Mathematical Modelling and Numerical Analysis*, 24(5), pp. 613–626.
- Ciarlet, P. (1979) The finite element method for elliptic problem. *Studies in Mathematics and its Applications*. North-Holland.
- Coward , A.C., P.D. Killworth and J.R. Blundell (1994) Tests of a 2-grid world ocean model. *Journal of Geophysical Research-Oceans*, 99:C11, pp. 22725–22735.
- Curchister, E.N., M. Iskandarani and D.B Haidvogel (1998) A spectral element solution of the shallow-water equations on multiprocessor computers. *Journal of Atmospheric and Oceanic Technology*, 15, pp. 510–521.
- Deleersnijder, E., J.P. van Ypersele and J.M. Campin (1993) An orthogonal, curvilinear coordinate system for a world ocean model. *Ocean Modelling*, 100, pp. 7–20.
- Eby, M. and G. Holloway (1994) Grid transformation for incorporating the arctic in a global ocean model. *Climate Dynamics*, 10, pp. 241–247.
- Foreman, M.G.G. (1984) A Two-Dimensional Dispersion Analysis of Selected Methods for Solving the Linearized Shallow Water Equations. *Journal of Computational Physics*, 56, 287-323.
- Fortune, S. (1987) A sweepline algorithm for voronoi diagram. *Algorithmica*, 2, pp. 153–174.
- Haidvogel, D.B., J.L. Wilkin and R. Young (1991) A semi-spectral primitive equation ocean circulation model using vertical sigma and orthogonal curvilinear horizontal coordinates. *Journal of Computational Physics*, 94, pp. 151–185.
- Iskandarani, M. and D. Haidvogel (1995) A staggered spectral element model with application to the oceanic shallow water equations. *International Journal for Numerical Methods in Fluids*, 20, pp. 393–414.
- Jayaraman, V., H.S. Udaykumar and W.S. Shyy (1997) Adaptive unstructured grid for three-dimensional interface representation. *Numerical Heat Transfer*, 32(3), pp. 247–265.
- Henry R.F. and R.A. Walters (1993) Geometrically based, automatic generator for irregular triangular networks. *Communications in Numerical Methods in Engineering*, 9 (7), pp. 555-566.

- Johnson, C. and A. Szepessy (1995) Adaptive finite element methods for conservations laws based on a posteriori error estimates. *Communications on Pure and Applied Mathematics*, XLVIII, pp. 199–234.
- LeProvost C., M.L. Genco, F. Lyard, P. Vincent and P. Canceil (1994) Spectroscopy of the World Ocean Tides from a Finite-Element Hydrodynamic Model. *Journal of Geophysical Research-Oceans*, 99 (C12), pp. 24777-24797.
- Le Roux, D.Y., A. Staniforth, C.A. Lin (1998) Finite elements for shallow-water equation ocean models. *Monthly Weather Review*, 126:7, pp. 1931–1951.
- Lynch, D.R. and F. Werner (1991) Three-dimensional hydrodynamics on finite elements. part II: Non-linear time-stepping model. *International Journal for Numerical Methods in Fluids*, 12, pp. 507–533.
- Lynch, D.R., J.T. Ip, C.E. Naimie and F.E. Werner (1996) Comprehensive coastal model with application to the gulf of maine. *Continental Shelf Research*, 16 (7), pp. 875–906.
- Madec, G. and M. Imbard (1996) A global ocean mesh to overcome the north pole singularity. *Climate Dynamics*, 12, pp. 381–388.
- Murray, R. (1996) Explicit generation of orthogonal grids for ocean models. *Journal of Computational Physics*, 126, pp. 251–273.
- Myers, P. and A. Weaver (1995) A diagnostic barotropic finite-element ocean circulation model. *Journal of Atmospheric and Oceanic Technology*, 12, pp. 511–526.
- Nielson, G. (1993) Scattered data modeling. *IEEE Computer Graphics and Applications*, 13-1, pp. 60–70.
- O’Rourke, J. (1993) *Computational geometry in C*. Cambridge University Press.
- Purser, R. and M. Rančić (1997) Conformal octagon: an attractive framework for global models offering quasi-uniform regional enhancement of resolution. *Meteorology and Atmospheric Physics*, 62, pp. 33–48.
- Renka, R. (1997) Algorithm 772: Stripack: Delaunay triangulation and voronoi diagram on the surface of a sphere. *ACM Transactions on Mathematical Software*, 23 (3), pp. 416–434.
- Shum C.K., P.L. Woodworth, O.B. Andersen, G.D. Egbert, O. Francis, C. King, S.M. Klosko, C. LeProvost, X. Li, J.M. Molines, M.E. Parke, R.D. Ray, M.G. Schlax, D. Stammer, C.C. Tierney, P. Vincent and C.I. Wunsch (1997) Accuracy assessment of recent ocean tide models. *Journal of Geophysical Research-Oceans*, 102 (C11), pp.

Watson, D. (1981) Computing the n-dimensional delaunay tessellation with applications to voronoi polytopes. *Computer Journal*, 24 (2), pp. 167–172.

Wilkin, J. and K. Hedström (1998) User's manual for an orthogonal curvilinear grid-generation package. Technical report.

Williamson, D. (1979) Difference approximation for fluid flow on a sphere. in: *Numerical methods used in atmospheric models (vol. II)*. GARP Publication Series n°17. WMO-ICSU.