A REVIEW OF THE DELAUNAY MESH GENERATION FOR HEAT TRANSFER FINITE ELEMENT ANALYSIS

Aluisio Viais Pantaleão
aluisio@mec.ita.br
Cláudia Regina de Andrade
claudia@mec.ita.br
Edson Luiz Zaparoli
zaparoli@mec.ita.br
Instituto Tecnológico de Aeronáutica – Departamento de Energia – 12228-900 – São José dos Campos, SP, Brasil.

Abstract. *A domain must be decomposed in sufficiently small elements to attain the desired accuracy in a numerical solution of partial differential equations that simulate a physical problem. In this work an unstructured mesh of triangular three nodes elements is generated automatically by a computer code that uses the Delaunay algorithm. To generate the first mesh, a set of points is distributed in the domain internal region according to user specified local mesh density parameter. To improve the iterative solution convergence rate of the algebraic equations system, a Delaunay algorithm is used to found the set of three vertices (mesh nodes) of each triangle maximizing the smallest internal angle of all mesh elements. In the triangular mesh obtained with the Delaunay algorithm, the circle whose circumference contains the triangle vertices is void, that is, no another mesh node is inside this circle. When the application of the mesh refinement error estimator over obtained results is higher than an user specified value, additional points are inserted in the more critical regions. The meshes obtained by this procedure are regular with nearly six triangle connected to each node and improve the overall numerical solution process.*

Keys word: *mesh generation, Delaunay Triangulation, Voronoi Diagrams, Heat Transfer*

1. Introduction

With the rapid development of numerical techniques for the engineering problems solution, there was an increasing demand for the automatic generation of computational meshes in the past few decades. In this context, the finite element method has emerged as one method for the solution of mathematical physics complex problems.

The computational mesh generation can be defined as the subdivision process of physical domain in smaller sub-domain, with the purpose to make easier the interpolation function inside each element and reduce the error in the numerical solution of the differential equation.

The importance of numerical solution technique is increasing in the design, simulations and analysis of complex problem of engineering, mainly in automotive and aerospace areas. Thus, the improvements in the robustness, velocity and quality of automatic mesh generator become of great importance in the process of dissemination of computational methods.

In the finite element method, the original domain can be subdivided in simple geometric forms (elements), such as: triangle and/or quadrilaterals to two-dimensional or tetrahedra, hexahedra and pentahedra to three-dimensional problems. Algorithms of the automatic mesh generation process may define the distribution and shape of the elements.

The problem of mesh generation is a delicate process, because consist in the determination of the number of nodes and their locations and elements (groups of nodes), with variables sizes and shapes that result in a better description of original geometric domain, but without committing the convergence of the numerical solution (Canann et al., 2000).

In the structured mesh each node pertains to a fixed number of elements or there are a fixed number of neighborhood nodes to each mesh node. However, in the unstructured mesh the number of elements that contains a mesh node is variable, thus there are a variable number of neighborhood nodes to each mesh node.

One of the first and well-succeeded attempts to generate finite element mesh with some degree of automation and mathematical consistency was due to Zienkiewicz and Phillips (1971).

The methods to construct unstructured mesh are frequently based upon geometrical ideas. There are several methods available such as Voronoi diagrams and Delaunay triangulation, advancing front technique and octree / quadtree. A survey of fundamental geometric data structure of the Voronoi diagrams and Delaunay triangulation is presented by Aurenhammer (1991). Their work demonstrates the importance and usefulness of the Voronoi diagram in a wide variety of fields inside and outside computer science and surveys the history of its development.

Mavriplis (1992) developed an algorithm for generating an unstructured mesh in an arbitrary two-dimensional configuration. That article combines the mathematical feature of Delaunay triangulation algorithm with the desirable point placement feature, boundary integrity, and robustness of the advancing-front-type mesh generation strategies.

In 1993, Rebay published a work that describes an unstructured mesh generation method entirely based on the Delaunay Triangulation. Rebay (1993) used the Bowyer-Watson algorithm to generate non-uniform mesh in domain of arbitrary shape. The method shown is computationally efficient and applicable both to two and three dimensions.

Rourke (1994) describes the construction of the Voronoi Diagrams with some definitions and basic properties. Their book presents the code to construct the dual of the Voronoi sites (Delaunay Triangulation).

Subramanian et al. (1995) developed an algorithm for two and three-dimensional automatic structured mesh generation. The algorithm was designed utilizing the scheme suggested by Zienkiewicz and Phillips (1971). Several examples are presented illustrating the effectiveness of the algorithm.

A new method for construction the Delaunay triangle computing the angles of points with respect to a baseline in a region instead of checking the empty circumcircle was proposed by Du (1996). Several examples and applications were included in his paper to demonstrate the efficiency of the algorithm for automatic Delaunay triangulation of arbitrary planar domains.

At this context, the present study shows the results of a computer code used to generate automatic unstructured mesh of triangular three nodes elements applying the Delaunay triangulation method coupled to the quadtree technique. Some examples are presented illustrating the performance of the used algorithm. A conduction heat transfer problem is solved to show the code mesh generation capability in an irregular domain.

2. Solution methodology: Galerkin finite element

At the present study, the Galerkin finite element method with automatic mesh generation was employed to obtain the solution of a conduction heat transfer problem. The finite element technique considers an approximated solution to a differential operator $L(\phi) - f = 0$, where f is the source-term. The approximated solution can be build with a linear sum of a finite set of basis functions as:

$$\tilde{\phi}(x,y) = \sum_{i=1}^{K} \phi_i \, W_i(x,y) \tag{1}$$

Here "K" is the number of nodes in domain $\Omega = \bigcup_e \Omega_e$ (union of the sub-domain of each element "e"), W_i is the basis functions or the global interpolation function that carry on the influence of the value of ϕ at node "i" over the domain. This approximated solution results in a residual that is calculated as:

$$\Re = L(\tilde{\phi}) - f \tag{2}$$

In the Galerkin Weighted Residual Method (Huang and Usmani, 1994), the nodal values ϕ_i are calculated by the algebraic equations system obtained when the residual is minimized, that is, or when each scalar product of the residual ($\Re$) with the basis functions (W_i) is reduced to zero as:

$$\int_{\Omega} (\Re \, W_i) \, dV = 0 \qquad i = 1, 2, \ldots, k \tag{3}$$

The global function (W_i) depends only of space coordinates and is defined inside each element "e" that has node "i" as one vertices by the elemental interpolation function W_i^e. The W_i^e functions have the following properties:

(a) they are zero out of the element "e";
(b) they assume unitary value in the node "i" and
(c) they assume null value in another node of element "e".

The global function W_i is obtained supposing that inside each element that contain the node "i" the function W_i is equal the W_i^e:

$$W_i = \bigcup w_i^e \text{ for all "e" elements that contains node "i"} \tag{4}$$

In the discretization of domain $\Omega = \bigcup_e \Omega_e$ a set of three nodes triangular elements will be used (Fig. 1).

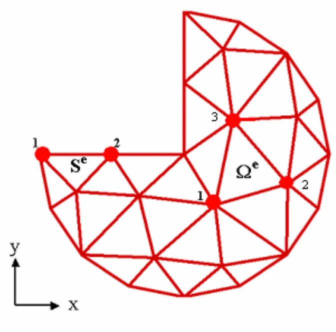

Figure 1. A two-dimensional domain discretized with triangular element.

A two-dimensional conduction heat transfer problem with constant properties can be represented by:

$$k\left(\frac{d^2T}{dx^2}+\frac{d^2T}{dy^2}\right)+Q=0 \tag{5}$$

where k is the thermal conductivity; T is the temperature field and Q is the volumetric heat generation rate.

Applying the Galerkin Weighted Residual Method (described in the previous section) and the Green Theorem to reduce the order of derivatives in diffusive term of the heat conduction differential equation (Eq. 5) the following equation is obtained:

$$\sum_{e=1}^{M}\int_{\Omega_e}\left[\frac{\partial W_i}{\partial x}k\frac{\partial \tilde{T}}{\partial x}+\frac{\partial W_i}{\partial y}k\frac{\partial \tilde{T}}{\partial y}\right]dx\,dy-\sum_{e=1}^{M}\int_{\Omega_e}W_i\,Q\,dxdy+\sum_{e=1}^{M}\int_{S^e}W_i\,\bar{q}\,dS=0 \tag{6}$$

Here $\tilde{T}$ represents the temperature field approximation inside element "e".

$$\tilde{T}^e(x,y)=\sum_{\Omega_e}^{i=1,P}T_i^e\,W_i^e(x,y) \tag{7}$$

where S_e indicate the surface of element "e" that has intersection with the boundary of domain Ω and "q" is the S_e normal component of heat flux vector, T_i^e is the temperature at each node of the element "e", P is number of nodes of element "e" and M is the total number of elements in Ω domain.

3. Automatic mesh generation

Several techniques for unstructured mesh generation were proposed in the past years. The problems of unstructured mesh generation is largely one of designing an algorithm that is automatic, robust, and yield suitable elements shapes and distributions for the flow and heat transfer solver. In Mavriplis (1997) a survey of techniques for generation of unstructured mesh is presented, such as: advancing-front technique, Delaunay triangulation, edge and face swapping technique, quad/octree-based methods, stretched mesh generation, mixed-element meshes. Of the techniques presented in Mavriplis (1997) two are suitable for unstructured mesh generation: the advancing-front method, and Delaunay-based approaches. While the advancing-front method is somewhat heuristic in nature, Delaunay-based methods are firmly rooted in computational geometry principles.

This paper solves numerically the conduction heat transfer problem (Eq. 5) and presents a description of a computer program that uses the quadtree technique to insert inner nodes within a 2D domain and the Delaunay triangulation to connect the nodes for each triangular element.

3.1 Quadtree based method

The quadtree is a hieraechical tree structure, which is based on the recursive subdivision of cells into four smaller of equal sizes (Yiu et al, 1996).

With this method, rectangles containing the geometric model are recursively subdivided until the desired resolution is reached. Figure 2 shows the two-dimensional quadtree decomposition of a model. Irregular cells are then created where rectangles intersect the surface, often requiring a significant number of surface intersection calculations. The quadtree technique does not match a pre-defined surface mesh, as an advancing front or Delaunay mesh might, rather surface facets are formed wherever the internal quadtree structure intersects the boundary. The resulting mesh also will change as the orientation of the rectangles in the quadtree structure is changed and can also require. To ensure element sizes do not change too dramatically, a maximum difference in quadtree subdivision level between adjacent rectangles can be limited to one. Smoothing and cleanup operations can also be employed to improve element shapes.

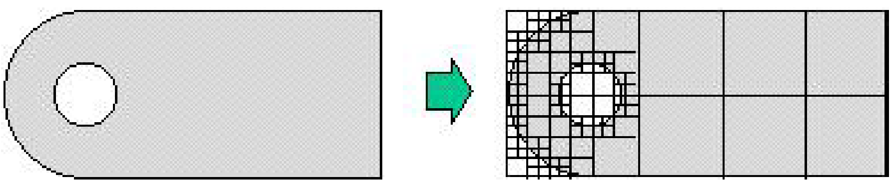

Figure 2. The two-dimensional quadtree decomposition of a model (Owen (2002)).

3.2 Mathematical features of theVoronoi diagram and Delaunay triangulation

Given a set of points in the plane, there exist many possible triangulations of these points. A Delaunay construction, that is the dual of the Voronoi diagram, represents a unique triangulation of these points with a large class of well-defined properties. Some properties can be employed to construct algorithms for generating the Delaunay triangulation of a given set of points.

Let $S=\{p_1, p_2, ..., p_n\}$ of $n \geq 3$ be a set of points in the two-dimensional Euclidean plane, thus a region can be assigned to each points p_i in S , such that the points in this region is closer to p_i than to any other point of S . These n regions divide the domain into a non-overlapping convex set known as the Voronoi diagram. All those points assigned to p_i form the Voronoi region $V(p_i)$. Mathematically, let $\{p_i\}$ denote a set S of n points, the Voronoi region $V(p_i)$ can be defined as:

$$V(p_i) = \left\{p : \|p - p_i\| \leq \|p - p_j\|, \ \forall j \neq i\right\} \tag{6}$$

Note that this defined set closed. Some points do not have a unique nearest point, or nearest neighbor. The set of all points that have more than one nearest neighbor form the Voronoi diagram edges for the set of p_i points (dashed lines in Fig 3).

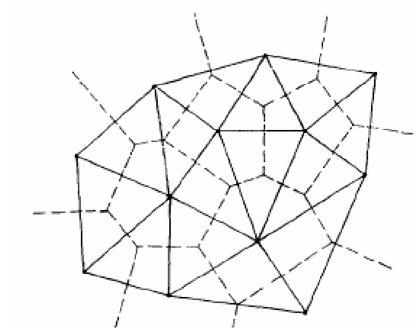

Figure 3. Voronoi diagram (dashed lines) and Delaunay triangulation (solid lines) on a set of points (Du (1996)).

In 1934 Delaunay (Rourke, 1994) demonstrated that when a dual graph is drawn with straight lines normal to each boundary of all Voronoi region, it produces a planar triangulation of the Voronoi diagram of *S* (if no four points are co-circular), called the Delaunay Triangulation. To obtain the Delaunay triangulation, each par of points p_i that share an edge of an Voronoi polygon is joined by a straight-line segment, thus resulting in a triangulation of the original n points set, represented in Fig. 3 by the solid lines.

Each vertex of the Voronoi diagram is located at the point of contact of three adjacent polygons and, also, defines the circumcentre for a Delaunay triangle. It is thus clear this if a triangle $\Delta p_1 p_2 p_3$ constructed by the straight-line segments connecting three pairs of points of S ($\{p_1p_2\}$, $\{p_2p_3\}$ e $\{p_3p_1\}$) satisfies the Delaunay criterion, that is, circumcircle does not contain any other point of S. Therefore, these given points will form a Delaunay triangle, if and only if the circumcircle defined by these points contains no other points in its interior, Fig. 4.

Delaunay triangulation has further properties (Du (1996)):

a) Assume that p_1p_2 is a Delaunay edge with end points p_1 and p_2, and $p_k \in S(k \neq 1,2)$ lies on one side of the edge p_1p_2. Let $\alpha_k = < p_1p_kp_3$ $(k \neq 1,2)$ be the interior angle of point p_k with respect to the edge p_1p_2, and there is a point 3 so that $\alpha_3 = < p_1p_3p_2$ = max $\{\alpha_{k,}$ $(k \neq 1,2)$ $\}$, where $p_3 \in S\{p_k, K \neq 1,2\}$, then the triangle $\Delta p_1 p_2 p_3$ is Delaunay satisfying (see Fig. 4).

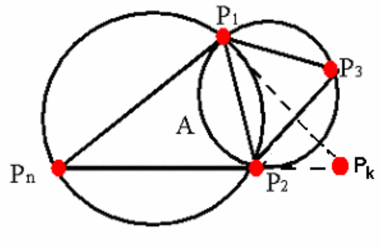

Figure 4. Verification of the Delaunay criterion.

b) The nearest neighbor p_j of a point p_i in the plane defines a Delaunay edge p_jp_i (for example, points 5 and 6 in Fig. 5).

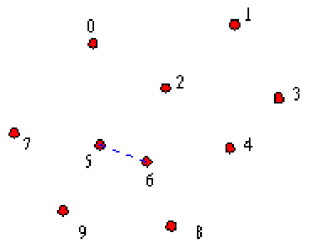

Figure 5. Point 5 nearest neighbor point 6 in the p_5p_6 edge of one Delaunay triangle.

c) Every edge of two adjacent points on the original boundary is a Delaunay edge (Fig. 6).

5. Mesh generation program

The program is divided in the following steps:

(a) **Geometric input:** This section is responsible by reading of the original boundary points of S, that are input in a counterclockwise order from any initial boundary point (Fig. 6). When this input sequence is used, interior points are located to the left of all boundary edges.

(b)

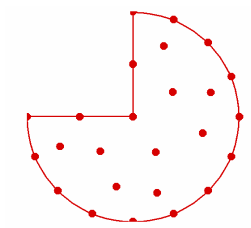

Figure 6. Data set and its original boundary.

(b) **Internal point creation:** A quadtree based method, Thompson et al. (1999) is used to generate points within the 2-D domain according to local mesh density information, which may be user defined or obtained from an error based adaptive analysis. The points for all domain is generate based on the given density distribution. At the first stage these density is imposed in an input file, where the values are defined in conformity to the necessary initial mesh refinement.

(c) **Construction of triangles**: Once the boundary points are defined and the inner points are added to the data set, the area can be triangulated. The set of internal points generated in the above step are connected by Watson's algorithm in three nodded triangles that satisfies the Delaunay criterion. Sloan and Houlsby (1984) describe an implementation of Watson's algorithm for computing two-dimensional Delaunay triangulations. In the present study, the code generates a table containing the point's coordinates (Tab.1) and the connectivity matrix (Tab. 2).

Table 1 Point's coordinates

Node	Coordinate x	Coordinate y
1	0.92	0.38
2	0.73	0.23
3	1.00	0.00
4	0.92	-0.38
5	0.66	-0.16
.	.	.
.	.	.
.	.	.

Table 2 Connectivity matrix

Triangle	Element nodes		
1	20	21	23
2	21	24	23
3	24	25	23
4	16	21	20
5	22	19	18
.	.	.	.
.	.	.	.
.	.	.	.

In the connectivity matrix (Tab. 2) each element is defined through 3 nodes identified by the global number whose coordinates are presented in the Tab. 1.

6. Mesh generation results

Using the Brazil's map, several examples showing the mesh features obtained by the above-described code are presented in Fig. 7 to 9, where the quadtree tecnique was implemented (Yiu et al, 1996). To generate a hollow region with this code, the nodes should be inserted in clockwise order. A mesh for a domain with an inside hole is presented in Fig.9. Figure 7 presents a domain with 566 nodes and 1010 elements unstructured nearly uniform mesh. This mesh was obtained with only one mesh density parameter.

Figure 7. Unstructured nearly uniform mesh in Brazil map.

Figure 8 presents an unstructured mesh with 2223 nodes and 4156 elements. This code allows mesh refinement in regions according to the adaptive solver requirements or by the user (the region of more intense gradients requires a mesh refinement attain the precision target). Two different mesh density parameters were used: one in the central region and another in the map extreme north and south regions. So, the code generates a mesh with a smoother transition between the remaining regions.

Figure 8. Top and bottom refined unstructured mesh inside the Brazil map.

Figure 9 presents Brazil's map without the Goias State, simulating a hole or cavity in the domain. This example uses a mesh with 2160 nodes and 4070 elements with unequal mesh refinement.

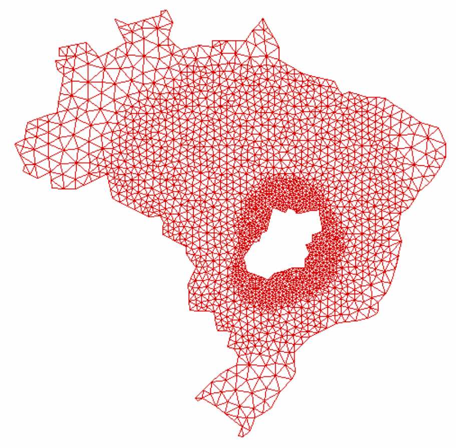

Figure 9. Triangulation of the Brazil's map with an internal hole.

In the previous example it was defined a gradual refinement from the cavity to the external boundary.

8. A two-dimensional conduction heat transfer results

Again, the Brazil's map was used to simulate the steady state two-dimensional conduction heat transfer (Eq. (5) with Q = 0). An arbitrary value of 100 [W / (m^2 K)] was used to the thermal conductivity parameter. In the map contour three thermal boundary conditions were imposed: a constant temperature equal to 400 K was imposed in the A-B segment of the north area, a temperature equal to 300 K was specified in the C-D segment of the south area (black segment in Fig. 10), and an insulated thermal condition was implemented in the remaining boundaries (B-C and D-E segments). This case shows an unstructured mesh with 2056 nodes and 3959 elements obtained with constant density parameter.

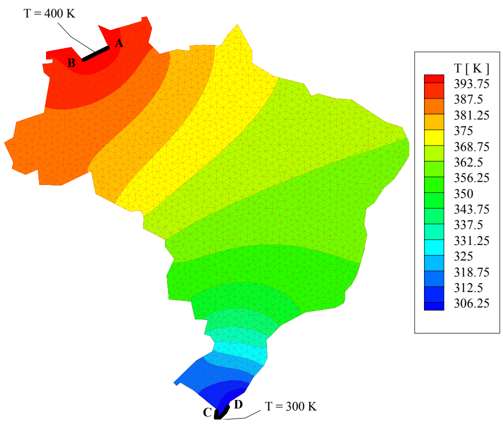

Figure 10. A two-dimensional conduction heat transfer.

The domain represented by the Brazil's map was chosen to test the code capability to generating a mesh in a complex geometry domain and solving the heat conduction problem. Results presented in Fig. 10, show the heat diffusion mechanism features with the end of the constant-temperature contours normal to the boundary domain where the insulated thermal condition was imposed. Heat diffuses from the high temperature region to the lower ones, taking account the heat transfer area enlargement.

7. Conclusions

Some features of a mesh generation algorithm based on Delaunay triangulation have been reviewed and a description of a quadtree/Delaunay code was presented. The computational code showed that the mesh obtained has good characteristics for finite element numerical heat transfer application. In spite of non-regular domain boundary and variable refinement, the generated meshes have nearly isosceles triangles. In this code the user could control the location and the mesh refinement intensity. Using an unstructured mesh mapping an irregular domain, the code showed a good performance to solve a conduction heat transfer problem.

8. Acknowledgements.

The first author acknowledges the support from CNPq (#146405/99-4) grant.

9. References

Aurenhammer, F. 1991, Voronoi Diagrams – A Survey of a Fundamental Geometric Data Struture", ACM Computing Surveys, vol. 23, n° 3, September

Canann, Scott A., Saigal S. and Owen, S. J., 2000."Special Edition on Unstructured Mesh Generation", International Journal for Numerical Methods in Engineering", Wiley, vol 1, 49.

Du, Chongjiang, 1996, "An Algorithm for Automatic Delaunay Triangulation of Arbitrary Planar Domains", Advances in Enginnering Software, 27, pp. 21-26.

Huang, H.C, and Usmani, A. S., 1994, "Finite Element Analysis for Heat Transfer", Springer-Verlag, New York.

Mavriplis, D. J., 1992, "An Advancing Front Delaunay triangulation Algorithm Designed for Roubstiness", Jounal of Computational Physics, 177, pp. 99-107.

Mavriplis, D. J., 1997, "Unstructured Grid Tecniques", Annu. Rev. Fluid. Mech, 29, pp. 473-514.

Owen, Steve, "Mesh Generation: A Quick Introdution", in www.andrew.cmu.edu/user/sowen/mintro.html, 2003.

Rebay, S., 1993, "Efficient Unstructured Mesh Generation by Means of Delaunay Triangulation and Boywer-Watson Algorithm", Journal of Computaional Physics, 106, pp. 125-138.

Rourke, O'. Joseph, 1994, "Computational Geometry in C.", Cambridge University Press,

Sloan, S. W. and Houlsby, G. T., 1984, "An Implementation of Watson's Algorithm for Computing 2-Dimensional Delaunay Triangulations". Advances in Engineering Software (1978), Volume 6 (4), pp 192-197.

Thompson, J. F, Soni, B. K., Weatherill, N.P.,1999, "Handbook of Grid Generation", CRC Press, New York.

Yiu, K. F. C., Greaves, D. M., Cruz, Saalehi, S. A. and Borthwick, A. G. L., 1996, "Quadtree grid generation: information handling, boundary fitting and CFD applications, Computers & Fluids, Volume 25 (8), pp 759-769.

Zienkiewicz, O. C. and Phillips, D. V., 1971, "An Automatic Mesh Generation Scheme for Plane and Surfaces by Isoparametric Coordinates". Int. J. Numer. Meth. Engng, 3, pp. 519-528.