Unstructured Mesh Generation with Delaunay Triangulation and Mesh Refinement with Local Spacing Control

Sutthisak Phongthanapanich
Graduate Student

Pramote Dechaumphai
Professor

Mechanical Engineering Department, Chulalongkorn University
Bangkok 10330, Thailand.

Abstract

This paper describes the logic of a dynamic algorithm for constructing two-dimensional triangular meshes using the Delaunay triangulation with automatic adaptive refinement feature. The complexity of the geometry both simply connected and multi-boundaries domains are completely arbitrary. Laplacian smoothing technique is applied to further improve the shape and size of triangular meshes. A corresponding computer program has been developed and evaluated by a number of problems with complex geometry.

1. Introduction

The finite element application requires discretization of the domain over which a set of governing equations is to be solved. Because of arbitrarily shape of domain, improved general-purpose mesh generation algorithms have been still in great demand. Because of varieties arbitrarily shape of domain, the unstructured mesh can provide multiscale resolution and conformity to complex geometries comparing to the structured mesh.

On the unstructured mesh approach, the two methods have proved particularly successful and are widely used today. Firstly, the advancing front method, the triangles are built progressively inward from the boundaries of domain until the domain area is filled with triangles [1]. Secondly, the Delaunay triangulation method, the popular meshing technique [2] that utilizes the Delaunay criterion. The Delaunay criterion in itself, is not a meshing technique. It provides the criteria for which to connect a set of existing points in space, normally are boundary points. Therefore, the point creating technique is required in addition, to refine the triangles. The refinement technique presented in this paper, follows the Weatherill and Hassan approach [3] which is widely used in engineering applications. The triangle aspect ratios are improved by applying the Laplacian smoothing technique [7] that moves each node of triangles to the centroid of all triangles around the node.

To demonstrate the advantages of the method proposed, this paper first describes the concept behind the Delaunay triangulation. The mesh generation procedure is then proposed with automatic point creation procedure. The Laplacian smoothing technique is then described to perform mesh smoothing. A number of complex geometries are then used to evaluate the capacity and effectiveness of the proposed method.

2. Delaunay Triangulation

2.1 Concept

Dirichlet [5-6] proposed a method to construct Dirichlet tessellation or Voronoi diagram, where as a domain could be decomposed into a set of packed convex triangles. For a given set of points in space, $\{P_k\}$, k=1,.... n, the regions $\{V_k\}$, k=1,...,m, are the boundaries which can be assigned to each point P_k, represent the space closer to P_k than to any other points in the set. Therefore, these regions satisfy,

$$P_k = \{P_i : |p - P_i| < |p - P_j|, \forall j \neq i\} \tag{1}$$

If all the points which have some segment of a Voronoi boundary in common are joined, the result is a Delaunay triangulation. In Graph theory, Delaunay triangulation could be defined that the graph which any circle in the plane is said to be empty if it contains no vertex in its interior [4]. This defining characteristic of Delaunay triangles, in Fig. 1, is called the empty circumcircle property.

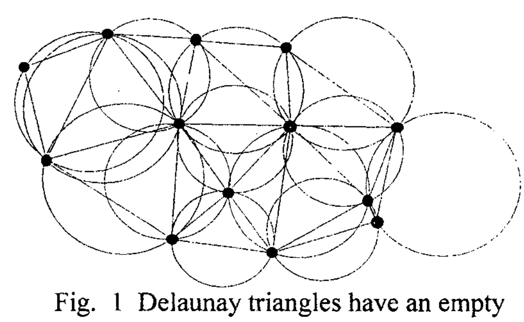

Fig. 1 Delaunay triangles have an empty circumcircle property.

2.2 Mesh Generation Procedure

The prior works that brought Delaunay triangulation into practical were introduced by Bowyer [5] and Watson [6] called Bowyer/ Watson algorithm. In this algorithm, when a new vertex is inserted, each triangle whose circumcircle contains the new vertex is no longer Delaunay (in-circle criterion) and is thus deleted all other triangles remain Delaunay are left undisturbed. Each vertex of the insertion polyhedron is then connected to the new vertex creating a new edge.

The algorithm used to generate Delaunay triangulation has two steps for two-dimensional domain. Firstly, forming triangles by connected points on the boundaries of domain called boundary triangle generation. Secondly, creating points inside domain to refine triangles of previous step to conform our desired in both shape and size.

The Delaunay triangulation algorithm that used to construct boundary triangle in this paper is based on the in-circle criterion according to Bowyer. The algorithm is described as the algorithm I below.

Algorithm I; DelaunayTriangulation(P, T)

```
Let P, k = 1,..., n, is the set of points on boundary of domain the stored in sequence of counter-clockwise direction;
Let T is the set of Delaunay triangles;
P.Initialize;
T. Initialize;
P.ReadDataFromStorage;
P.DefineConvexHull;
P.AddPoint(p1, p2, p3, p4);
T.AddTriangle(t1, p1, p2, p3);
T.AddTriangle(t2, p2, p3, p4);
for i=1 to n {
    t ← T.FindTriangleContainPoint(P(i));
    T0 ← T.IncircleTriangle(t);
    T ← T0.CreateNewTriangle;
    T.AssignNeighborhoodTriangle;
};
T.RemoveOutsideDomainTriangle;
end;
```

Delaunay triangles of the boundary points of domain that constructed by algorithm I are shown in Fig. 2. The left figure shows the plate with circular cutout domain and boundary point. The middle figure shows boundary Delaunay triangles and right figure shows boundary Delaunay triangles after deleting triangles outside the domain. In this figure, all triangles inside circular cutout domain are thus deleted. To determine which triangle is outside the domain, the cross product of vectors of two contiguous sides of triangle is less than zero or ordering of vertices of triangles is clockwise rather than counter-clockwise basis.

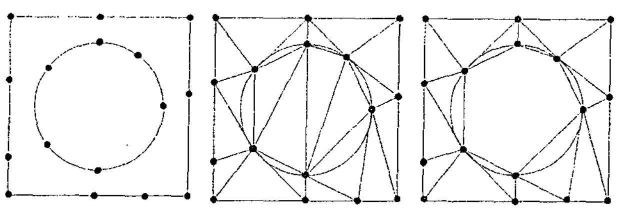

Fig. 2 Delaunay triangles of the plate with circular cutout domain with Algorithm I.

2.3 Automatic Point Creation Procedure

The Delaunay triangulation algorithm that described above does not suggest how to create points inside the domain. Many researchers introduced approaches how to create points inside the domain, to refine boundary triangles that number of methods use the set of boundary points to guide point placement [7-11, 14-15]. The automatic point creation procedure in this paper derived from the algorithm suggested by Weatherill [3] and Karamete [12].

The shape and size of triangles or density of points inside domain that created by this scheme control by two coefficients. Alpha coefficient controls point density by changing the allowable shape of formed triangles and Beta coefficient controls the regularity of the triangulation by not allowing point within a specified distance of each other to be inserted in the same sweep of the triangles within the field. The combination of Alpha and Beta coefficients cause shape and size triangles varies. The suggested values of Alpha and Beta coefficients for coarse and refine triangular mesh are 0.8 and 0.9, and 0.5 and 0.6 respectively.

The main idea of automatic point creation scheme is searching triangle that accepts Alpha and Beta criteria testing and placing new point at the centroid of the triangle. Then new triangles could be created by Delaunay triangulation algorithm that described in algorithm I. The detailed implementation of automatic point creation scheme are described in algorithm II.

Algorithm II; MeshRefinement(P, T, alpha, beta)

```
Let P, k = 1,..., n, is the set of points on boundary of domain the stored in sequence of counter-clockwise direction;
Let T is the set of Delaunay triangles;
Let Alpha is the coefficient that controls shape of formed triangles;
Let Beta is the coefficient that regularity of the triangulation;
do t ← T {
    p ← t.ComputeTriangleCentroid();
    dp ← t.ComputePointDistribution();
    for j = 1 to 3 {
    If (t.CentroidDistance(p, j) < alpha * dp) break;
    };
    P0 ← T.FindInsertedPointInNearestTriangle;
    do p1 ← P0 {
        If (distance(p, p1) < beta * dp) break;
    };
    P.AddPoint(p);
    DelaunayTriangulation(P, T);
};
end;
```

The demonstration of a new point creation inside domain with automatic point creation scheme is shown in Fig. 3. The new point that passes both Alpha and Beta criteria testing is inserted at the centroid of shaded triangle. In-circle criteria testing is applied to all neighborhood triangles. Finally, new triangles are formed with Delaunay triangulation algorithm.

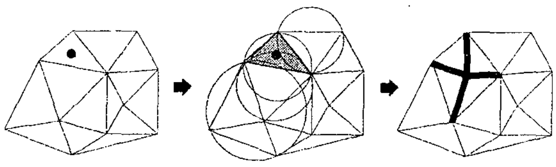

Fig. 3 Mesh refinement with automatic point creation scheme (Algorithm II).

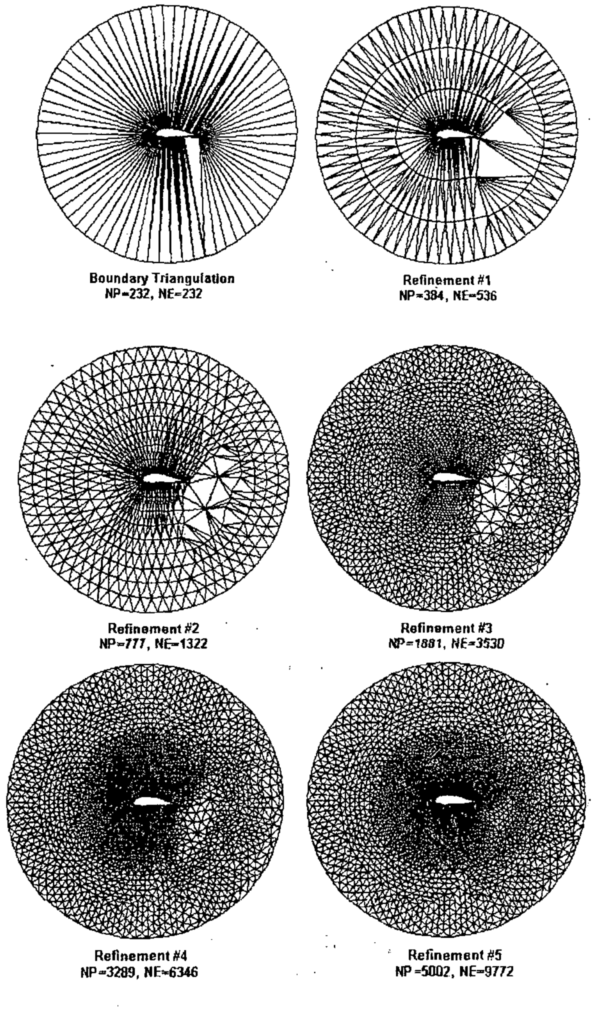

Fig. 4 Mesh refinement with automatic point creation scheme, α=0.5 and β=0.6.

3. Mesh Smoothing

Shape and size of triangles created from previous steps can be improved by applying the Laplacian smoothing technique. The points repositioning formula can be derived as a finite difference approximation of Laplace's equation [7, 13]. Each interior point is moved successively to the centroid of its connected neighborhood points that the points of all neighborhood triangles. Several passes are made through the entire set of interior points to obtain the optimized shape and size triangles.

$$x_{ic} = \frac{\sum_{i=1}^{M} x_i}{M} \quad and \quad y_{ic} = \frac{\sum_{i=1}^{M} y_i}{M} \qquad i = 1,2,\ldots,M \tag{2}$$

Number of times to applied mesh smoothing technique is arbitrarily. Normally, there are two ways to decide when to stop mesh smoothing procedure. The first way is to assign number of smoothing times manually by user. The second way is to assign maximum point movement distance. The mesh smoothing procedure stops when there is no point movement distance greater than the given value. The sample figures in this paper use value 0.001 as mesh smoothing stopping criterion.

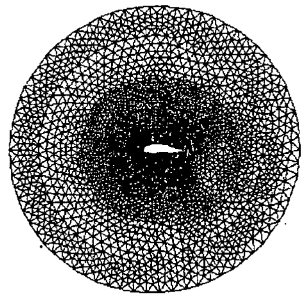

Fig. 5 Mesh improvement with Laplacian smoothing technique.

4. Examples

Five examples are used to evaluate the capacity and effectiveness of the Delaunay triangulation technique. Triangular meshes constructed by the proposed method has been applied to many fields such as heat transfer, solid mechanics, low-speed and high-speed fluid flows. Figure 6 shows triangular mesh of plate with circular cutout for conduction heat transfer analysis. Figure 7 shows triangular mesh of cutter blade for stress-strain analysis. Figure 8 shows triangular mesh of two-floor house for low Reynolds number computational fluid dynamics analysis. Figure 9 shows triangular mesh of fighter aircraft and Fig. 10 shows triangular mesh of shuttle and fuel tank for high-speed flow computational fluid dynamics analysis.

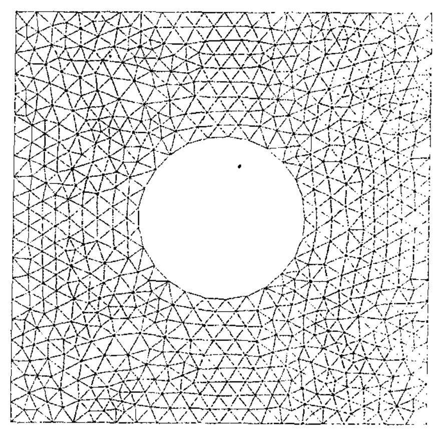

Fig. 6 Triangular mesh for plate with circular cutout.

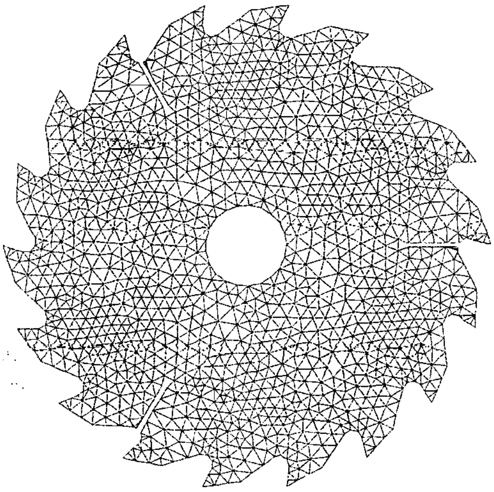

Fig. 7 Triangular mesh of a cutter blade.

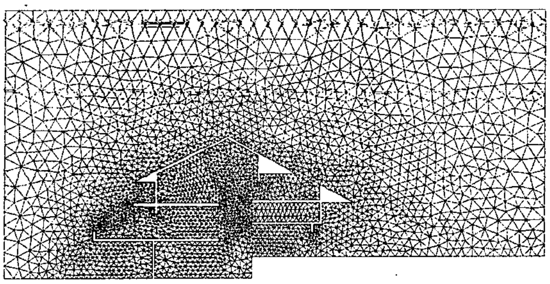

Fig. 8 Triangular mesh for analysis of flow past a house.

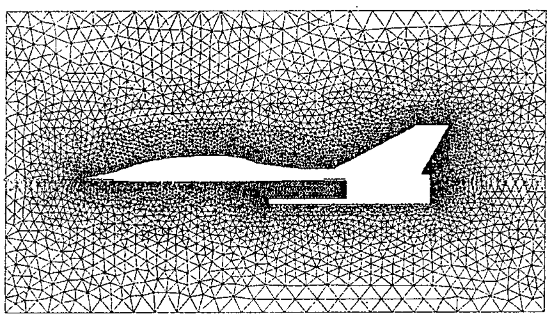

Fig. 9 Triangular mesh for flow past a fighter aircraft.

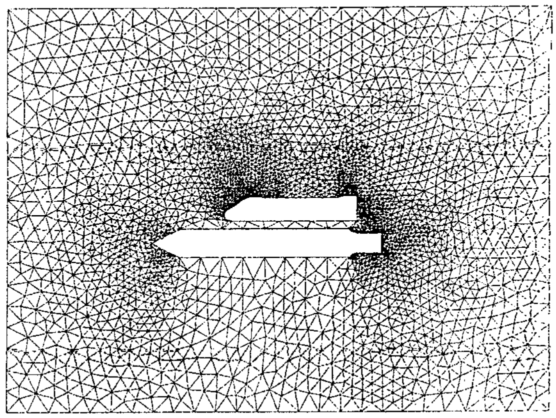

Fig. 10 Triangular mesh for flow past a shuttle.

5. Conclusion

This paper has discussed a method to construct unstructured mesh of triangles based on Delaunay triangulation. Mesh refinement algorithm has been suggested by automatic point creation scheme. Several examples have demonstrated the capacity of the method for generating effective meshes for finite element analysis.

References

[1] Lo, S. H., "A New Mesh Generation Scheme for Arbitrary Planar Domains", *International Journal for Numerical Methods in Engineering*, Vol. 21, 1985, pp. 1403-1426.

[2] Owen, S. J., "A Survey of Unstructured Mesh Generation Technology", Department of Civil and Environmental Engineering, Carnegie Mellon University, Pittsburgh, 1998.

[3] Weatherill, N. P. and Hassan, O., "Efficient Three-Dimension Delaunay Triangulation with Automatic Point Creation and Imposed Boundary Constraints", *International Journal for Numerical Methods in Engineering*, Vol. 37, 1994, pp. 2005-2039.

[4] Chew, L. P., "Constrained Delaunay Triangulations", *Algorithmica*, Vol. 4, 1989, pp. 97-108.

[5] Bowyer, A., "Computing Dirichlet Tessellations", *The Computer Journal*, Vol. 24, No. 2, 1981, pp. 162-166.

[6] Watson, D. F., "Computing the n-Dimensional Delaunay Tessellation with Application to Voronoi Polytopes", *The Computer Journal*, Vol. 24, No. 2, 1981, pp. 167-172.

[7] Frey, W. H., "Selective Refinement: A New Strategy for Automatic Node Placement in Graded Triangular Meshes", *International Journal for Numerical Methods in Engineering*, Vol. 24, 1987, pp. 2183-2200.

[8] Nambiar, R. V., Valera, R. S. and Lawrence, K. L., "An Algorithm for Adaptive Refinement of Triangular Element Meshes", *International Journal for Numerical Methods in Engineering*, Vol. 36, 1993, pp. 499-509.

[9] Ruppert, J., "A Delaunay Refinement Algorithm for Quality 2-Dimensional Mesh Generation", *Journal of Algorithms*, Vol. 8, No. 3, 1995, pp. 548-585.

[10] Joe, B., "Delaunay Triangular Meshes in Convex Polygons", *SIAM Journal on Scientific and Statistical Computing*, Vol. 7, No. 2, 1986, pp. 514-539.

[11] Joe, B. and Simpson, R. B., " Triangular Meshes for Regions of Complicated Shape", *International Journal for Numerical Methods in Engineering*, Vol. 23, 1986, pp. 751-778.

[12] Karamete, B. K., Tokdemir, T. and Ger, M., "Unstructured Grid Generation and A Simple Triangulation Algorithm for Arbitrary 2-D Geometries Using Object Oriented Programming", *International Journal for Numerical Methods in Engineering*, Vol. 40, 1997, pp. 251-268.

[13] Cavendish, J. C., "Automatic Triangulation of Arbitrary Planar Domains for the Finite Element Method", *International Journal for Numerical Methods in Engineering*, Vol. 8, 1974, pp. 679-696.

[14] Lo, S. H., "Delaunay Triangulation of Non-Convex Planar Domains", *International Journal for Numerical Methods in Engineering*, Vol. 28, 1989, pp. 2695-2707.

[15] Jin, H. and Wiberg, N. E., "Two-Dimensional Mesh Generation, Adaptive Remeshing and Refinement", *International Journal for Numerical Methods in Engineering*, Vol. 29, 1990, pp. 1501-1526.

[16] Shewchuk, J. R., "Triangle: Engineering a 2D Quality Mesh Generator and Delaunay Triangulator", School of Computer Science, Carnegie Mellon University, Pittsburgh, 1997.