Memoirs of the Faculty of Engineering, Okayama University, Vol.25, No. 2, pp.123-133, March 1991

Mesh Generation for Convex 3-Dimensional Domain

Takeo TANIGUCHI* and Chikashi OHTA**

(Received February 8 , 1991)

SYNOPSIS

The aim of this investigation is the proposal of 3D mesh generation method based on the Delaunay triangulation. The method is valid for the finite element modelling of any convex 3D domain into tetrahedra with optimum geometrical configuration. This paper includes the mathematical background of the mesh generation method, its detail, proposal of some efficient tools for faster and more rigorous computations, and some examples of the mesh generation.

1. INTRODUCTION

The mesh generation method determines the utility of the finite element method, and effective tools have been developed in recent years. Especially, in case of 3 dimensional problems effective mesh generator is inevitable at the application of FEM.

Preferable 3D mesh generator should be

(1) fully automated,

(2) reliable for the results,

(3) the one which can give good finite elements,

(4) applicable for arbitrary node distribution, and

(5) fast.

---

* : Department of Engineering Science

** : Department of Civil Engineering

At present we have two basic ideas for 3D mesh generation, which are the Octree method and the Delaunay triangulation. The comparison of their functions clarifies that the latter is superior in forming finite elements under the conditions of (3) and (4). [1, 2, 3]

The Delaunay triangulation is the dual problem of the Voronoi tesselation, and they can fill the domain occupied by distributed data points by using tetrahedra and convex polyhedra, respectively. Each polyhedron generated by the Voronoi tesselation includes a data point, and each polygon on the surface of the polyhedron locates at the middle plane between neighbouring two data points. Then, four data points which are separated by three intersecting polygons on a surface of the polyhedron locate on a same circumsphere of the tetrahedron formed by these four data points. If every four among all data points are selected to form tetrahedron whose circumsphere does not include any other data point within it, the tetrahedra necessarily shows good geometrical configuration.

The purpose of this investigation is to give an algorithm of the Delaunay triangulation which can be applied for 3D space. For this purpose the authors firstly give several mathmatical theorems on the geometry of the Voronoi tesselation and the Delaunauy triangulation, and the results are directly used for the mesh generation method of 3-dimensional domain. A number of useful tools are also given for the fast and rigorous computation and the generalization of the method. Proposed method is the direct application of the Delaunay triangulation, and it can be applied only to any 3-dimensional domain with convex boundary configuration.

2. MATHEMATICAL BACKGROUND OF 3D MESH GENERATION

Suppose the positions of n distinct points are given, and we assume that the Delaunay triangulations are completed for these data points. Then, each circumsphere does not include any other data points inside the sphere. Our aim is to find new Delaunay triangulation occured by the addition of new data point. A part of following mathematical results are already proved by T.J. Baker.[4] For the simplicity we use following notations.

Tet(ABCD) : A tetrahedron decided by nodes, A, B, C and D.

Sph(ABCD) : A circumsphere of Tet(ABCD)

Tri(ABC) : A triangle decided by nodes, A, B, and C.

#1 The tetrahedra whose circumspheres include the additional point P are adjacent each other through faces.

[Proof]

Assume that a data point P newly set in the domain locates in Tet(ABCD). Then, Sph(ABCD) obviously includes P.

Here, we assume that P is also included in Sph(IJKL) of another tetrahedron Tet(IJKL) which is not adjacent to Tet(ABCD). Then, Sph(IJKL) must penetrate all tetrahedra locating between Tet(ABCD) and Tet(IJKL) without including any data points which construct these penetrated tetrahedra.

Let one of these tetrahedra penetrated by Sph(IJKL) be Tet(EFGH). Then,

$$\mathrm{Sph(IJKL)} \cap \mathrm{Sph(EFGH)} = \emptyset$$

as shown in Fig.1. From the geometrical relation between Tet(ABCD), Tet(EFGH) and Tet(IJKL), P must be located in Sph(IJKL) Sph(EFGH). Then, P must be included in Sph(EFGH).

Above relation between Sph(EFGH) and Sph(IJKL) is valid for all tetrahedra locating between Tet(ABCD) and Tet(IJKL), and P is included in all of their circumspheres. Then, all of these tetrahedra are connected each other, since their faces are penetrated by Sph(IJKL).

Since the tetrahedra obtain in #1 are connected each other through trianglar faces, they form a polyhedron by removing all triangles which locate between them.

#2. The polyhedron of #1 can be filled by tetrahedra each of which is formed by using three nodes of a triangle on the surface of the polyhedron and the new data point P.

[Proof]

It is obvious that the surface of the polyhedron is covered by triangles. Then, the proof completes if three nodes of any triangle on the polyhedr-

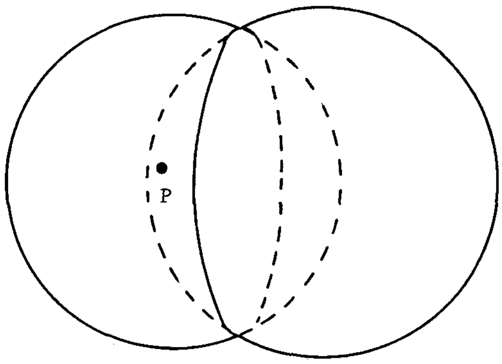

Fig.1 Relation between Two Circumspheres

on can be connected with P by straight lines.

The polyhedron is originally filled by tetrahedra whose circumspheres include P. Pick up a tetrahedron among them. Then, all nodes of the tetrahedron locate on the surface of its circumsphere, and P locates inside of the sphere. Then, P can be connected with these nodes by straight lines. This logic is valid for all tetrahedra constructing the polyhedron.

#3. The tetrahedra obtained in #2 is the Delaunay triangulation for n+1 nodes.

[Proof]

Firstly we prove that the tetrahedra in the polyhedron obtained in #2 is the Delaunay triangulation. If P is not used for forming new tetrahedra in the polyhedron, then new tetrahedra must be formed by using data points on the surface of the polyhedron. But, their circumspheres obviously include other nodes (for example, P) inside of it. Then, one node of any tetrahedron newly formed in the polyhedron must be P.

Successively, we show that new Delaunay triangulation for the polyhedron never give influence to the construction of tetrahedra outside of the polyhedron. Consider a pair of tetrahedra which locate inside and outside of the polyhedron and face each other. Let these two tetrahedra be Tet(ABCP) and Tet(ABCD). Then, two nodes, P and D, locate on opposite side of Tri(ABC), because Tet(ABCD) is a Delaunay triangulation and P locates inside of the polyhedron. If we assume that Sph(ABCP) includes the point D, P must be also included in Sph(ABCD). This shows the contradiction of the assumption that Sph(ABCD) never includes P. Then, the part of Sph(ABCP) outside of the polyhedron must be included within Sph(ABCD), and it never includes D.

Above mathematical results indicate the modification method of the Delaunay triangulations due to the addition of another data point in the domain filled by tetrahedra in the sense of the Delaunay triangulation. That is, if the Delaynay triangulation is obtained for a set of data points, new Delaunay triangulation after adding another data point into the domain can be obtained by the triangulation of the subdomain which is occupied by the tetrahedra whose circumspheres include the added point. And, new tirangulations complete by connecting all data points on the surface of the subdomain to dew data point P.

4. MESH GENERATION OF 3D DOMAIN

4.1 Preparation of 3D Mesh Generation

Three mathematical results in previous section directly lead to the mesh generation procedure of 3-dimensional space as following: Assume that a number of data points are set and the domain occupied by these points are divided into tetrahedra of the Delaunay triangulation. And, we give another data point in the domain at arbitrary position. Then, following steps can lead to new Delaunay triangulation.

Step 1. Find circumspheres which include the new data point.

Step 2. Remove all the triangles which locate between adjacent tetrahedra, and form a polyhedron.

Step 3. Form new tetrahedra by using all triangles on the surface of the polyhedron to the new data point.

Above procedure is the main part of the Delaunay triangulation, and we consider on its realization as the algorithm.

(1) *Scaling of the domain occupied by data points.*

Data points are arbitrarily distributed in 3D space, and the size of the domain along x, y, and z axes depends on the problem. In order to use the Delaunay triangulation as a general-purpose tool we introduce the scaling of the domain. For this purpose, we find the maximum length of the domain along x, y and z axes, and divide the lengthes along three directions by the maximum value.

(2) Setting of Supertetrahedron

The mathematical procedure of the Delaunay triangulation given in this section is the repetition of partial modification of the subdomain filled by tetrahedra. Then, the tetrahedron which includes new data point must be, at least, modified.

Set an imaginary tetrahedron which can enclose whole of the domain occupied by data points, and start to divide the domain into tetrahedra by giving data points one by one. The setting of the first data point into this imaginary tetrahedron necessarily divides it into four smaller tetrahedra. Same subdivision of a tetrahedron into four may occure at the set of successive data points.

As obvious from above explanation the introduction of the imaginary tetrahedron can simplify and standardize the procedure of the Delaunay triangulation. We call this imaginary tetrahedron the super-tetrahedron.

(3) Searching of the tetrahedron including new data point

Those tetrahedra whose circumspheres include new data point must be found at every setting of new data point. For this purpose we propose following method.

Let Tet(ABCD) be a tetrahedron including new data point. Then, the summation of the volume of four small tetrahedra which are constructed by the set of the data point in Tet(ABCD) is equal to the volume of Tet(ABCD). But, it is time-consuming directly to introduce this method for finding a tetrahedron including new data point, because it requires numerous number of volume calculations. In order to decrease the number of repetitions of this searching procedure, we begin the searching from the tetrahedra which are generated at the last stage.

Assume that we could find out the tetrahedron which includes new data point. Then, for finding the tetrahedra whose circunspheres include the data point, we use following prosedure: By using the adjacency relationship between generated tetrahedra and the tree-searching technique we find such tetrahedron that the square of its radius is larger than the square of the distance between the circum-center and the data point.

(4) Treatment of Degeneracy

The generacy is the case where several optimum tetrahedra simultaneously occure at the setting of new data point. [2,3] In our method the final state of the subdivision into tetrahedra by setting a data point is left for the successive stage.

4.2 Input and Output Data

Assume that we treat n data points which are distributed in 3D space. Then, the input data are 1) the number of data points (NODE), and 2) their x, y, and z coordinates (PX, PY, PZ, respectively).

After the Delaunay triangulation we have to prepare the data for the finite element analysis which are 1) the number of tetrahedra, 2) the element-node relations, and 3) the coordinates of data points. The last item is given as input data. For giving the boundary condition of the finite element analyssis we are often required the nodes locating on the surfaces of the domain. This data is easily obtained from the data generated through this procedure.

4.3 Mesh Generation Procedure

[Step 1] Input of data

Prepare the data of NODE, PX, PY, and PZ, and input them.

[Step 2] Setting of Supertetrahedron

Calculate the maximum size along x, y and z axes of the domain occupied by all of data points, and reset the coordinates of all data points so that the maximum size is equal to 1. And, set the supertetrahedron so that the data points occupy its central area. Here, we set the radius of the supertetrahedron to be 10.

[Step 3] Setting of a data point and Searching the tetrahedra which must be modified by the addition of the point.

Pick up a data point, and find those tetrahedra whose circunspheres include the point. This search consists of two methods which are explained in previous section. These tetrahedra form only one polyhedron by removing all common triangles which locate between two adjacent tetrahedra.

[Step 4] Triangulation of the polyhedron

The polyhedron obtained in Step 3 is triangulated by using triangles on the surface of the polyhedron and new data point.

(Steps 3 and 4 are repeated till all of the data points are introduced for the triangulation.)

[Step 5] Removal of all tetrahedra which include the forming points of the supertetrahedron

Among the generated tetrahedra we remove all tetrahedra which include the points forming the supertetrahedron, and the residual tetrahedra are the necessary finite elements.

4.4 Recognition of the results

Proposed method is based on the mathematical investigation and it must be rigorous theoretically, but the occurance of numerical error cann't be overcome as far as some numerical operations are introduced into the procedure. Since some errors may occure through the mesh generation process, we have to prepare how to examine the result of the mesh generation. For this purpose we propose following methods:

(1) The examination whether the generated tetrahedra are optimum.

This examination can be done by surveying whether each cirsumsphere does not include any other data point inside of the sphere.

(2) The examination whether the domain is filled by tetrahedra.

If a pair of tetrahedra are adjacent each other, same triangle appears in different two tetrahedra. On the other hand, if a triangle can be found in only one tetrahedron, it must be the one on the surface of the domain. By gathering all triangles which appear only once, we can examine whether the domain is filled by tetrahedra. This examination method can be easily visualized by using the stereoscopic figures.

5. EXAMPLES OF 3D MESH GENERATION

The first example of the mesh generation is for the recognition of the dual relation between the Voronoi tesselation and the Delaunay triangulation. As shown in Fig.2 we place eight data points in 3D space, and the domain is divided into tetrahedra by using the proposed method. Fig.2 illustrates both of the Voronoi tesselation and the Delaunay triangulation using the stereoscopic figures. From the figure the domain is successfully divided into polyhedra of the Voronoi tesselation and also into tetrahedra of the Delaunay triangulation.

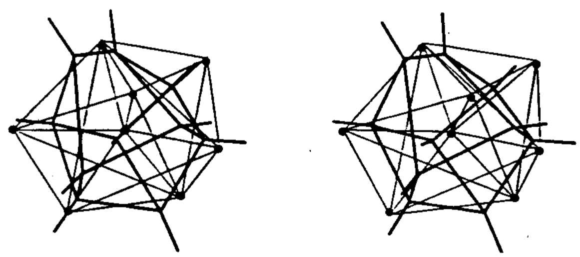

Fig.2 Voronoi Tesselation and Delaunay Triangulation

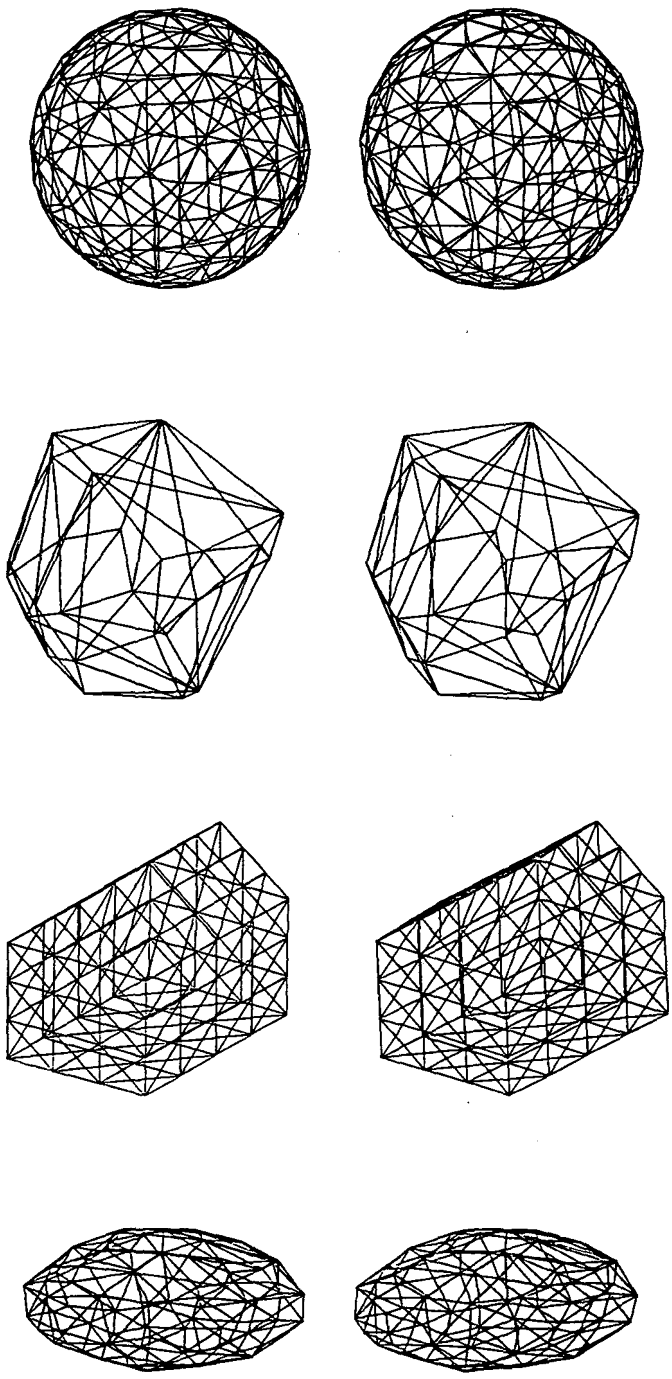

Fig.3 Delaunay Triangulation

Several figures of the Delaunay triangulation are also given in Fig.3. Since too many data points are placed in 3D space, these figures show only the triangulations on their surfaces. These figures can be good examples of the examination method given in Section 4.4.

30 problems are used for testing the proposed method, and the results are summarized as a graph which shows the relation between the required CPU time and the number of data points. (See Fig.4) This figure shows that the CPU time increases almost linearly in accorance with the number of data points, but for some cases the mesh generation procedure requires relatively long execution time. They are the cases that all data points are placed only on the surface of spheres, and, therefore, most of the execution time is consumed for the occurence of the degeneracy. That is, in case of sphere the tetrahedra covering the sphere have the same circumsphere (i.e. the original sphere), and every setting of new data point the phenomenon of the degeneracy occure. The data points are systematically generated for some test problems (* in Fig.4) and randomly done for the residuals (o in the graph). But, as obvious from the graph, the difference between these methods does not give serious influence to the CPU time. This shows the proposed method is effective for actual mesh generation method for 3D problem. We should note that these tests are done by using a 16-bit personal computer.

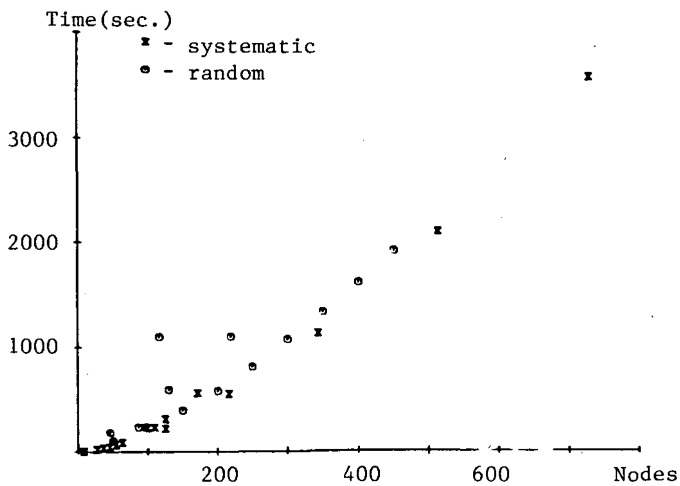

Fig.4 Execution Time for Mesh Generation

6. CONCLUDING REMARKS

In this investigation the authors proposed the Delaunay triangulation method for 3-dimensional domain, which is directly applied as the mesh generator for 3D finite element analysis.

Proposed method is based on the Voronoi teselation and the Delaunay triangulation, and, therefore, they require only the location of data points. Then, there exists no concept of "boundaries" of the domain, and we cann't introduce the geometrical property of the boundaries of the domain which is used for the finite element analysis. That is, the convexity on the surface of the domain cann't be recognized, and the convex subdomain is also filled by tetrahedra after the mesh generation. This suggests that the definition and the recognition of the concept of "boundaries" are necessary for the development of more general-purpose mesh generator based on the Delaunay triangulation.

REFERENCES

[1] M.A.Yerry & M.S.Shephard,"Automatic Three-dimensional Mesh Generation by the Modified Octree Technique", Int. J. Num. Meth. Eng., Vol.20, 1984, pp.1965-1990

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

[3] D.F. Watson,"Computing n-dimensional Delaunay Tesselation with Application to Voronoi Polytopes", The Computer Journal, Vol.24, No.2, 1981, pp.167-172

[4] T.J. Baker,"Generation of Tetrahedral Meshes around Complete Aircraft", Numerical Grid Generation in Computational Fluid Mechanics '88 (ed. S. Sengupta et al), 1988, pp.675-685