3-D Automatic Mesh Generation for FEA Using Dynamic Bubble System

Takayuki Yokoyama, Vlatko Cingoski, Kazufumi Kaneda and Hideo Yamashita
Faculty of Engineering, Hiroshima University, 1-4-1 Kagamiyama, Higashihiroshima 739-8527, Japan

Reprinted from
IEEE TRANSACTIONS ON MAGNETICS
Vol. 35, No. 3, May 1999

3-D Automatic Mesh Generation for FEA Using Dynamic Bubble System

Takayuki Yokoyama, Vlatko Čingoski, Kazufumi Kaneda and Hideo Yamashita
Faculty of Engineering, Hiroshima University, 1-4-1 Kagamiyama, Higashihiroshima 739-8527, JAPAN

***Abstract*—In this paper, an extension of the previously developed 2-D dynamic bubble system for 3-D automatic mesh generation using tetrahedral finite elements is presented. Initially, a set of vertices inside the entire analysis region is generated using 3-D dynamic bubble system, followed by automatic mesh generation according to the Delaunay tessellation algorithm and the generated set of vertices. The proposed method is highly robust and easily applicable to convex and concave analysis domains with various geometrical complexity. The proposed method provides high quality tetrahedral meshes with graded mesh densities utilizing very small amount of input data.**

***Index terms*—Finite element methods, automatic mesh generation, Delaunay triangulation, mesh quality.**

I. Introduction

The main computation power of the finite element method results from its fundamental idea of replacing a given unknown function by piecewise approximations over a set of geometrically simple domains called finite elements. However, although really ingenious, this idea requires meshing of the entire analysis domain which is usually very laborious and time consuming, especially in 3-D space. A large number of methods for automatic mesh generation have already been proposed with various success. On the other side, it is well known that mesh generation methods based on the Voronoi polygons and the Delaunay algorithm are best suited to FEA since they always minimize the maximum angle of the finite elements. In order to execute the Delaunay triangulation algorithm, first a set of nodes with good location properties inside the analysis domain must be generated, which is not usually an easy and quick task.

Recently, we developed and presented a very promising method for automatic mesh generation [1] based on the dynamic bubbles system in 2-D space [2]. In this paper, an extended version of this method for 3-D automatic mesh generation is presented. Due to the existence of multiply connected and convex shaped domains filled with multimaterial regions, several typical 3-D problems had appeared and had to be additionally solved. The developed method featured a robust and widely applicable automatic mesh generation algorithm, as a result of which high quality tetrahedral finite element meshes with graded mesh densities could be generated easily utilizing extremely small amount of input data.

Manuscript received June 3, 1998.

Hideo Yamashita, +81 824-24-7665, fax +81 824-22-7195, yama@eml.hiroshima-u.ac.jp, http://www.eml.hiroshima-u.ac.jp/

II. Proposed Meshing Method

A. Dynamic Bubble System

The dynamic bubble system consists of a set of bubbles in which each bubble is defined with its radius, mass, and position in the space according to its central coordinates. Each bubble obeys the Newton's Second Law of Dynamics:

$$m_i \frac{d^2 a_i}{dt^2} + c_i \frac{da_i}{dt} = f_{a_i}(d) \tag{1}$$

where m_i, is the mass of bubble i, c_i is the viscosity coefficient, a_i is the x, y, or z coordinate of the center of bubble i, and $f_{a_i}(d)$ is the force acting on bubble i along direction a [1].

Forces $f(d)$ acting among bubbles can be attractive or repulsive depending on positions and radii of neighboring bubbles. In general, these forces can be mathematically expressed by the van der Waals' forces acting between two charges in the space [1].

The simplified procedure for the proposed automatic mesh generation system is shown in Fig. 1. First, the outline of the entire analysis domain must be specified using CAD or other geometrical modeling method. Next, radii of several vertex bubbles, mainly those which outline the entire analysis domain or part of it such as single material domains, must be set by the user or automatically approximated using some approximation technique. This is the only part of the procedure that requires user's control and expertise. The main advantage of this method is the development of dense meshes around vertex bubbles with smaller radii, and the opposite, generation of coarse meshes in the neighborhood of the vertex of a larger bubble radius. After the initial radii of the vertex bubbles are set, the entire procedure is automatically executed. First, the generation of edge bubbles is performed and movement according to the existing dynamic forces among bubbles is performed, followed by generation and movement of the facet bubbles in the same manner as above. At the end, according to the previous procedures, volume bubbles are generated inside the entire analysis domain. When the dynamic stability of the entire generated dynamic system of bubbles is achieved, the movement stops and each center of an existing bubble becomes one vertex in the finite element mesh. Finally, a tetrahedral finite

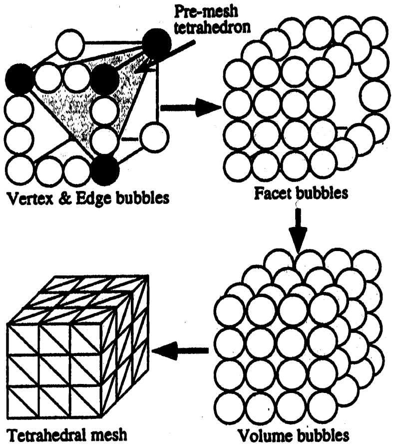

Fig. 1. Simplified procedure for automatic mesh generation using bubble dynamic system.

element mesh is generated utilizing the Delaunay algorithm [3] over the set of vertices generated above.

To avoid problems in 3-D mesh generation such as overlapping elements or meshing areas outside of the analysis domain, which mainly occurs due to the existence of concave and/or multimaterial analysis domains, initially the entire analysis domain is divided into a very coarse mesh, so called tetrahedral pre-mesh as shown in Fig. 1. This pre-mesh is generated utilizing the Delaunay tessellation method, using only vertices that outline the entire analysis domain.

B. Generation of Bubbles

For good and fast convergence of the bubble system dynamics, the initial positions and radii of bubbles have to be carefully determined. Since the generation of facet and volume bubbles is performed by the same algorithm, here, for simplicity we will explain only the generation of facet bubbles. The bubble generation procedure can be divided into two main procedures:

- definition of the initial bubble position, and
- computation of the radius of each bubble.

C. Definition of the Initial Bubble Position

The definition of the initial position of each facet (and/or volume) bubble is executed according to the following procedure:

- The entire analysis domain is divided into a very coarse tetrahedral mesh called pre-mesh as shown in Fig. 2a.
- Each boundary triangular facet of each pre-mesh tetrahedron is transformed from world coordinate system into X–Y plane as shown in Fig. 2b.
- the transformed boundary facet is mapped onto an area coordinate system L_2–L_3 in that manner that the vertex with the smallest radius of vertex bubble is always placed at the center of the coordinate system as shown in Fig. 2c.

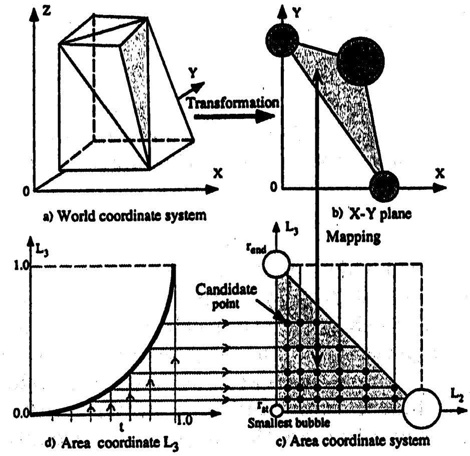

Fig. 2. Definition of candidate points for facet bubbles.

- The number and density of the generated bubbles is very important. If the number of bubbles is too low and they are set too coarse, a finite element mesh with low mesh quality will be generated. On the other side, if the number of bubbles is too big, and they are set too dense, a large computation time is needed to achieve the system's stability. In our method setting of the number and position of bubbles is executed in two steps: (1) Generation of candidate points inside the mapped triangular facet, and (2) decision on which of these candidate points will become a center of a new bubble.

For the generation of the candidate points, we used an exponential function for both local coordinates L_2 and L_3, respectively. The value of L_3 is computed according to the following equations:

$$L_3 = \frac{d^t - 1}{d - 1} \quad ; \tag{2}$$

$$d = \frac{r_{end}}{r_{st}} \quad ; \tag{3}$$

where r_{st} is the radius of a vertex bubble at the center of the coordinate system (the smallest radius), and r_{end} is the radius of the terminal vertex bubble along line L_3 (see Fig. 2c). The same procedure is performed for area coordinate L_2. Next, each cross sectional point which belongs to the mapped triangular surface becomes a candidate point for setting a new bubble.

However, not every candidate point becomes a new bubble. For each candidate point to become a new bubble the following inequality must be satisfied:

$$\beta \cdot (r_{can} + r_{old}) < d \quad ; \tag{4}$$

where r_{can} and r_{old} are the radii of the candidate bubble and the previously existing bubble in the neighborhood of the candidate point, respectively, while d is the distance between bubble's centers as shown in Fig. 3. If a candidate point satisfies the above inequality, then a new bubble is set at that point. Otherwise, the procedure continues with the next available candidate point until all candidate points are

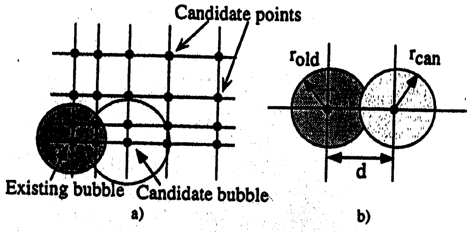

Fig. 3. Decision if a candidate point becomes a new facet bubble.

tested. The coefficient $0.5 < \beta < 1.0$ is defined by the user. Our experience shows that setting $\beta = 0.75$ almost always ensures generation of finite element meshes with high mesh quality at a reasonably fast computation time.

Finally, we would like to point out that setting the volume bubbles is executed by a simple extension of the above algorithm into the third dimension.

D. Computation of the Radius of Bubbles

The radius of each new bubble is interpolated using the radii of all vertex bubbles of the pre-mesh tetrahedron inside which this new bubble should appear as shown in Fig. 4. The interpolation is done according to the following exponential equation

$$r(L_1, L_2, L_3, L_4) = A\, e^{(B \cdot L_2 + C \cdot L_3 + D \cdot L_4)} \quad , \qquad (5)$$

where L_1, L_2, L_3, and L_4 are the volume coordinates, and the coefficients A, B, C, and D are computed using the radii of vertex bubbles $r_1 \sim r_4$ [1].

III. Mesh Generation Using Delaunay Tessellation Method

In the previous section we discussed the generation of a suitable set of vertices distributed into the 3-D space according to the dynamic bubble system. Next, we speak on how to generate a correct tetrahedral mesh division using this set of vertices, utilizing a Delaunay tessellation algorithm. The Delaunay algorithm works fine with a convex domain, but, it usually exhibits problem with subdivisions of concave domains. Additionally, meshing of multi-material domains can sometimes result in incorrect meshing. To solve these problems, we perform mesh generation for each object separately. However, in some cases such as that shown in Fig. 5a, the Delaunay tessellation can not be correctly executed without adding additional vertices, especially on the boundary planes. In our algorithm, the generation of additional vertices is enabled and a new vertex can be added either at the center of the fictitious triangle generated from three cross sectional vertices between the boundary plane and the edges of an incorrectly generated tetrahedron (see Fig. 5b), or at the center of the boundary edge between two boundary vertices of the incorrectly generated tetrahedron (see Fig. 5c).

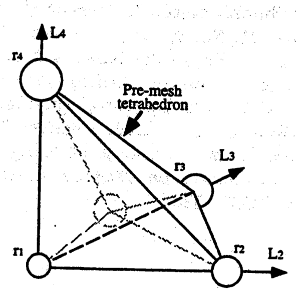

Fig. 4. Computation of radius of volume bubbles.

A. Mesh Quality

The mesh quality is very important factor in order to obtained an accurate finite element analysis. It is well known that equilateral triangles in 2-D and equilateral tetrahedrons in 3-D provides higher accuracy of the results than flat or slender triangles or tetrahedrons. Therefore, we used the values of the coefficient Q

$$Q = \frac{radius\ of\ inscribed\ sphere}{radius\ of\ circumscribed\ sphere} \times 3 \quad , \qquad (6)$$

as a geometrical mesh quality estimator. If the values of Q approach towards 1 the quality of the developed mesh is high and the shape of the generated tetrahedrons is almost equilateral. On the other side, if the coefficient Q approaches 0, the mesh quality is low and the generated tetrahedrons are flat and slender.

IV. Applications

The usefulness of the proposed 3-D automatic mesh generation method based on dynamic bubble system was verified on two examples. The first example is a model of a reactor which consists of a yoke and a square coil, for which the generated mesh is presented in Fig. 6. Figure 6a shows the entire mesh, while Figs. 6b and 6c show the generated meshes separately for the air domain and the yoke-coil domain. As can be seen, the developed mesh has smooth and graded mesh densities with high mesh quality. Meshing of the most difficult part, the very thin air gap regions, was done correctly and smoothly as can be seen from Fig. 6b.

The second example was a test model developed by the Japanese Institute of Electrical Engineers (JIEE). Generated finite element meshes for the entire analysis domain, air gap region and objects regions are shown in Figs. 7a, b and c, respectively. Similar to the reactor model, a mesh with high quality and graded mesh density can be observed. The shapes of the tetraheda generated around coil–ferrite–plate area are almost equilateral.

Regarding mesh quality, we checked the distribution of the mesh quality coefficient Q with the number of generated tetrahedrons for both models. Both models have

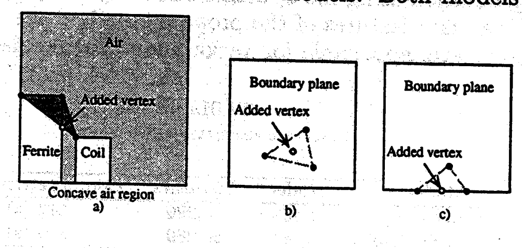

Fig. 5. Adding vertices for concave domains.

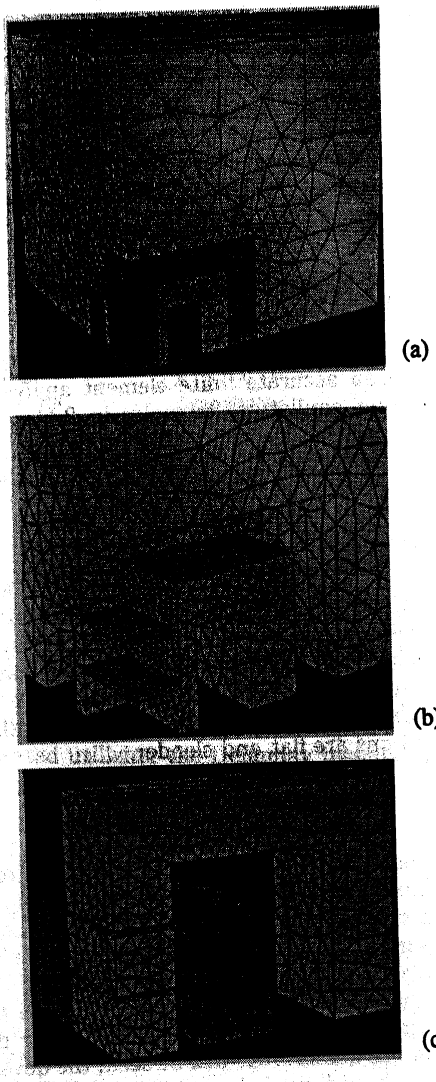

Fig. 6. Reactor model: (a) Final division map (b) Air region (c) Object region.

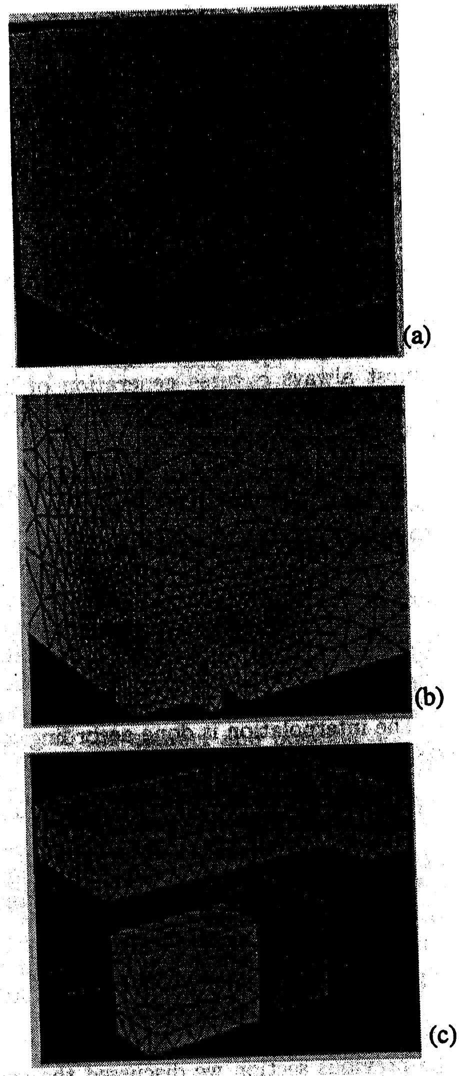

Fig. 7. Test model of the JIEE: (a) Final division map (b) Air region (c) Object region.

more than 80 % of the total number of generated finite elements with mesh quality coefficient larger than 0.7. The average values were 0.8 for the reactor model, and 0.82 for the JIEE test model. Table I, shows number of nodes, finite elements and a computation time for the generated finite element meshes using an SGI O2 Workstation.

V. Conclusions

In this paper, a new method for 3-D automatic mesh generation using the dynamic bubble system is presented. The main features of the proposed method are: small input data, applicable for various complex geometrical domains, generation of meshes with graded densities and high quality tetrahedra. As a future work, we would like to further improve the computation speed and to apply this meshing algorithm for 3-D adaptive mesh generation.

TABLE I

Generated finite element mesh data

Model	Nodes	Elements	Computation time
Reactor	2,867	15,296	142 (s)
JIEE Model	5,083	28,269	675 (s)

References

[1] V. Čingoski, R. Murakawa, K. Kaneda, and H. Yamashita, "Automatic mesh generation in finite element analysis using dynamic bubble system", *Journal of Applied Physics*, 15 April 1997, Vol. 81, No. 8, Part 2. pp.4085-4087.

[2] K.Shimada, "Physically-based automatic mesh generation," *Journal of Japanese Society for Simulation Technology*, Vol. 12, N. 11 (1993) pp.11-20 (in Japanese).

[3] D. N. Shenton and Z. J. Cendes, "Three-Dimensional Finite Element Mesh Generation Using Delaunay Tessellation", *IEEE Transaction on Magnetics*, November 1985, Vol. 21, No. 6, pp.2535-2538.