Studies in
Applied Electromagnetics and Mechanics

Volume 11

Vol. 7. C. Baumgartner, L. Deecke, G. Stroink and S.J. Williamson (Eds.), Biomagnetism: Fundamental Research and Clinical Applications
Vol. 8. R. Collins, W.D. Dover, J.R. Bowler and K. Miya (Eds.), Nondestructive Testing of Materials
Vol. 9. T. Honma (Ed.), Advanced Computational Electromagnetics
Vol. 10. A.J. Moses and A. Basak (Eds.), Nonlinear Electromagnetic Systems
Vol. 12. T. Takagi, J.R. Bowler and Y. Yoshida (Eds.), Electromagnetic Nondestructive Evaluation

Volumes 1-6 have been published by Elsevier Science under the series title "Elsevier Studies in Applied Electromagnetics in Materials".

ISSN: 1383-7281

Applied Electromagnetics and Computational Technology

Proceedings of the 4th Japan-Hungary Joint Seminar on Applied Electromagnetics in Materials and Computational Technology
Fukuyama, Japan, July 1-3, 1996

Edited by

H. Tsuboi
Department of Information Engineering
Fukuyama University, Japan

and

I. Sebestyen
Department of Electromagnetic Theory
Technical University of Budapest, Hungary

IOS Press
Ohmsha

Amsterdam • Berlin • Oxford • Tokyo • Washington, DC

Applied Electromagnetics and Computational Technology
H. Tsuboi and I. Sebesteyn (Eds.)
IOS Press, 1997

Automatic Mesh Generation Using Bubble System

Vlatko Čingoski, Seiki Noda, Ryo Murakawa, Kazufumi Kaneda and Hideo Yamashita
Faculty of Engineering, Hiroshima University
1-4-1 Kagamiyama, 739 Higashi-hiroshima, Japan

Abstract

A new method for automatic mesh generation using dynamic bubble system is presented. The proposed method has two separate routines: one which generates nodes inside the analysis domain using physically-based system of bubbles, and the second routine for automatic generation of finite elements according to the Delaunay algorithm making use of the previously generated set of nodes. Generation of the initial nodes in the analysis region is performed using frontal method while the dynamic movement of the bubbles is performed simultaneously. In the proposed method the density of the mesh can can be controlled using simple exponential functions which allow obtaining desired mesh density with modest amount of input data for a very short computation time. The proposed meshing method is applicable for an automatic meshing of complicated shapes and structures such as those usually found in various electromagnetic devices like rotating machines and transformers.

1. Introduction

Recently, with the tremendous progress in hardware and software technology, the finite element method became widely applicable for obtaining numerical solutions of differential equations commonly appearing in various engineering fields such as stress and strain analysis, electromagnetic and thermal field analysis, computation of fluid dynamics, etc. Its fundamental idea is to divide the domain under investigation into finite number of simple sub-domains and to approximate the unknown variable in each sub-domain by low order piecewise approximation functions - usually polynomials. Therefore, special attention must be paid to the subdivision of the entire analysis domain into a finite set of geometrically simple sub-domains called finite elements, a procedure usually known as finite element grid or mesh generation. The development of an optimal method for automatic mesh generation is of paramount importance and depends not only on the domain that has to be meshed but also on the type of analysis, the level of the desired accuracy of the results, the unknown variable and its distribution in the analysis domain as well as the capability of the available hardware support [1]. However, an optimal method for automatic mesh generation must unconditionally satisfy the following criteria [2]:

- To perform correct meshing with only a small and easily obtainable amount of input data;

- To generate finite elements which provide better accuracy for the same order of t approximation functions (e.g. in 2-D space triangles with equal lengths of edges
- To be easily applicable for meshing arbitrary complicated geometrical shapes a structures inside the analysis domain;
- To have short computation time.

In this paper, we propose a new method for automatic mesh generation of domai with complicated geometrical shapes using a physically-based model of dynamic bubbl A system of bubbles is a dynamic system of physical objects which have their own radi and mass and obey the Second Newton's Law of Dynamics [3]. According to their positi in space and the type and intensity of the forces acting among them, each bubble c move in the space until the stationary position of the entire system is obtained. In t same time, the radius of each bubble is automatically controlled according to its positi inside the analysis domain. After the moving process is finished, each center of a bubl becomes one node of the generated finite element mesh. Therefore, the meshing proce by the proposed method can be sumarized in the follwing two steps:

Step 1: — Generation of a set of nodes using dynamic bubble system, and

Step 2: — Construction of the finite element mesh using the Delaunay algorithm a the set of nodes previously generated in *Step 1*.

In what follows, the definition of the physically-based bubble system is given and t dynamic forces acting among bubbles are defined. Then, the method for generation of t bubbles together with their approximation functions is discussed and the mapping betwe arbitrarily shaped analysis space and simplified computational space is described. Finall the usefulness of the proposed method is verified using several test models.

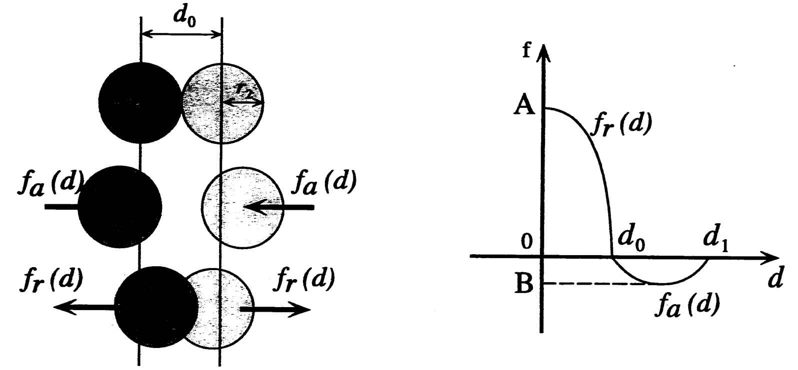
2. Physical properties of a dynamic bubble system

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

$$m_i \frac{\mathrm{d}^2 a_i}{\mathrm{d}t^2} + C\frac{\mathrm{d}a_i}{\mathrm{d}t} = f_{ia} \quad , \qquad (\!$$

where m_i is the mass of bubble i, C is the viscosity coefficient, a_i is the x, y or z coordina of the center of bubble i and f_{ia} is the force acting on bubble i along direction a. Forc $f(d)$ acting among bubbles are presented in Fig. 1. They can be attractive or repulsi forces. The character of the force f depends on the radius of the bubbles and their positic as shown in Fig. 1a. The mathematical representation of the force $f(d)$ is given with tl following equations

$$f(d) = \begin{cases} f_r(d) & 0 \le d < d_0 \\ 0 & d = d_0 \\ f_a(d) & d_0 < d \le d_1 \end{cases} \quad , \qquad (\!$$



a) Attractive $f_a(d)$ and repulsive $f_r(d)$ forces b) Force intensity distribution

Fig. 1 Definition of dynamic forces acting among bubbles.

where the distance d_0 is defined by the user as shown in Fig. 1. Partial forces $f_r(d)$ and $f_a(d)$ are defined as a second order polynomial functions. Together with their first derivatives $f_r'(d)$ and $f_a'(d)$ they must satisfy the following boundary conditions

$$\begin{array}{lll} f_r(0) = A & f_r(d_0) = 0 & f_r'(0) = 0 \\ f_a(d_1) = 0 & f_a(\frac{d_0 + d_1}{2}) = B & f_a'(\frac{d_0 + d_1}{2}) = 0 \end{array} \quad . \qquad (3)$$

3. Proposed algorithm for automatic mesh generation

One of the main advantages of the proposed method for automatic mesh generation based on dynamic bubble system is that it needs very small amount of input data. Therefore, it is very useful for users with limited knowledge of mesh generation or as a part of a large system for finite element analysis incorporated into wider CAD/CAM system. In the following, we describe the main algorithm for generation of the bubbles, definition of the mesh nodes and finally construction of the finite elements. For simplicity, in this paper we discuss automatic mesh generation in 2-D space only, although extension of the proposed algorithm in 3-D space is very straightforward.

In order to describe the proposed algorithm, a simple quadratic domain presented in Fig. 2 for automatic mesh generation is treated in this paper. Initially, user defines the radiuses of the initial bubbles, so called vertex bubbles, at several typical vertices inside the analysis region. Usually these vertices are the corners of the analysis domain as shown in Fig. 2. or corners between different materials. However, the program also allows setting additional vertex bubbles according to the desired mesh density at other specific positions inside the analysis region. By the rule, setting large radius for a vertex bubble means

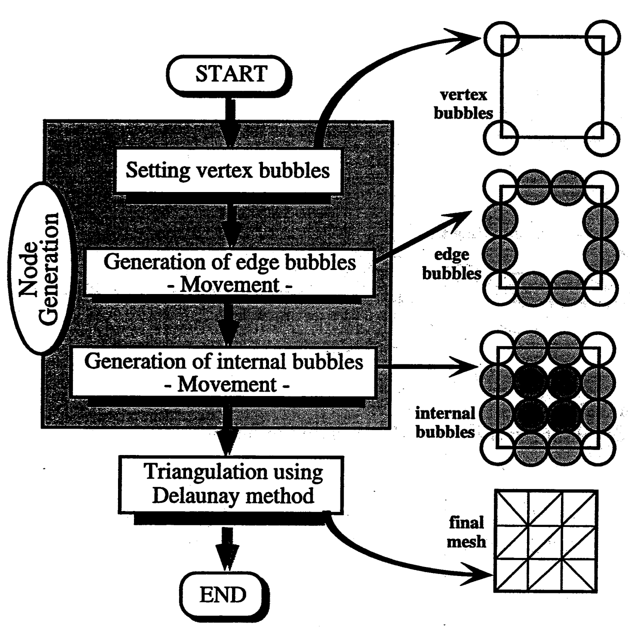


Fig. 2 Simplified algorithm for the proposed automatic mesh generation system.

development of a rough mesh in its neighborhood and opposite, setting small radius alwa results with dense mesh around that particular vertex. This method enables generati of a nearly desired mesh density with small amount of input data and flexible changes the mesh topology and density by simply changing the radiuses of several vertex bubble

After defining the radiuses of the vertex bubbles, the automatic generation of the bu bles begins. We used frontal method for the generation of initial bubbles, therefore, t program first generates the edge bubbles, and then internal bubbles (for 2-D space surfa bubbles, and for 3-D space volume bubbles). Next, according to the the dynamics of t system the movement of the bubbles is performed.

3.1. Generation of edge bubbles

Generation of edge bubbles depends on the radiuses of the vertex bubbles at bo terminal nodes of the edge r_1 and r_2. The number of generated bubbles n along one ed is computed using the average value of the radiuses of terminal vertex bubbles

$$N = \text{Nearest whole number of } \{\frac{l}{r_1 + r_2}\} \quad . \qquad ($$

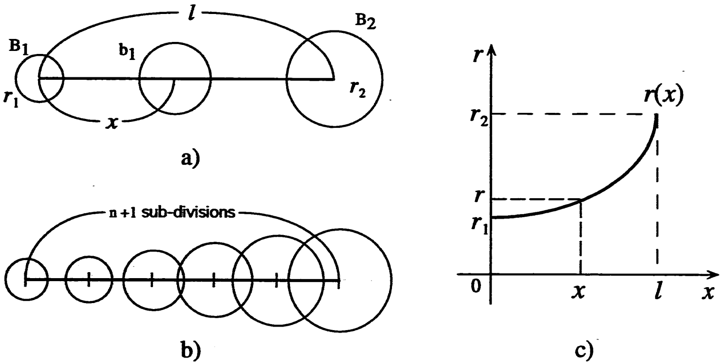


Fig. 3 Generation of edge bubbles ($n = 4$) : a) arbitrary edge bubble b) division of an edge on equal intervals $n + 1$ and generation of bubble c) exponential function $r(x)$.

Therefore, the length of the edge l is divided into $n + 1$ equal intervals if the computed number of nodes on this edge is n. Then, according to the previously defined exponential function

$$r(x) = A\, e^{Bx} \quad , \tag{5}$$

the radius of the bubble $r(x)$ at each point with coordinate x can be computed. The coefficients A and B in Eq. 5 can be easily computed using boundary conditions

$$r(0) = r_1 \qquad \text{and} \qquad r(l) = r_2 \quad . \tag{6}$$

Graphical representation of the edge bubble generation process is given in Fig. 3.

3.2. Generation of surface bubbles

For the generation of the surface bubbles similar method as the one described above is employed. Now, instead of only two terminal vertex bubbles, each surface can have three or more vertex bubbles. For simplicity the proposed algorithm incorporates two simple shapes of two-dimensional surfaces: a quadrilateral and a triangular, as shown in Fig. 4.

A. Quadrilateral surface: The arbitrary shaped quadrilateral surface can be meshed using the proposed method utilizing surface mapping procedure between the real surface (x, y) and the unit computational surface (u, v) (see Fig. 5a). The center of each initial bubble can be obtained using simple division procedure of the real two-dimensional space using the number of edge bubbles as shown in Fig. 5b for $n = 3$ and $m = 4$. Next, the initial radius of each surface bubble $r(u, v)$ inside the computational surface (u, v) is determined using the following interpolation function

$$r(u, v) = A\, e^{Bu+Cv+Duv} \quad , \tag{7}$$

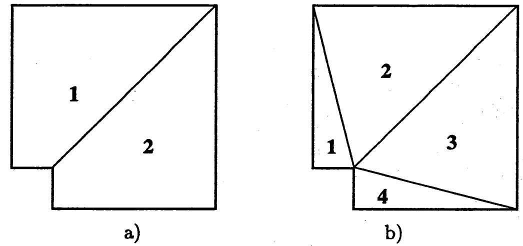


Fig. 4 Subdivision of an analysis domain: a) into quadrilateral surfaces b) into triangular surfaces.

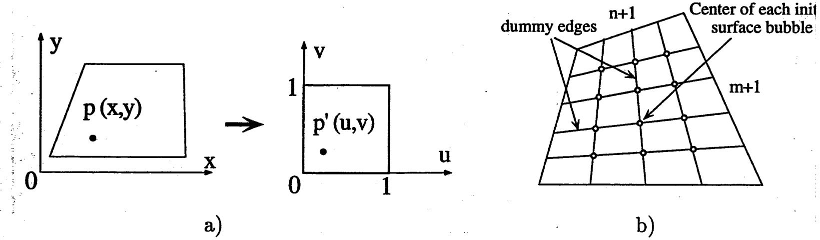


Fig. 5 Generation of surface bubbles: a) mapping between real surface (x, y) and uni computational surface (u, v) b) generation of initial surface bubbles ($n = 3$ and $m =$

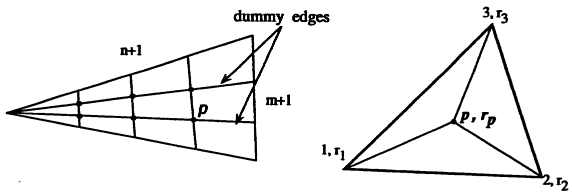
where coefficients A, B, C and D can be computed using the following boundary con tions

$$r(0,0) = r_1 = A \qquad r(0,1) = r_2 = A\, e^{C}$$
$$r(1,1) = r_3 = A\, e^{B+C+D} \qquad r(1,0) = r_4 = A\, e^{B} \quad .$$

B. Triangular surface: The radius interpolation for any surface bubble inside triangu surface is also performed using simple exponential function and the values of the radius of its terminal nodes. First, a set of dummy edges is constructed knowing the numb of divisions along each edge of the triangular surface and centers of initial bubbles placed at each cross point as shown in Fig. 6a. Next, the radius of each inital bubble at arbitrary point p inside the triangular surface (Fig. 6b) is computed using the surfa coordinates L_1, L_2 and L_3 and the following interpolation function

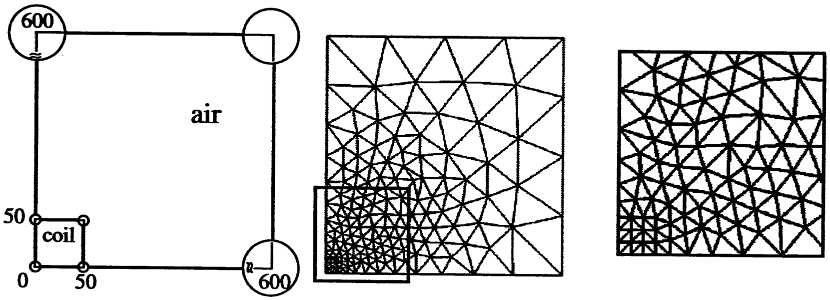
$$r(L_1, L_2, L_3) = A\, e^{BL_2+CL_3} \quad .$$

The coefficients A, B and C can be easily computed using the following boundary con tions



a) b)

Fig. 6 Generation of internal bubbles for triangular surfaces ($n = 2$ and $m = 3$).



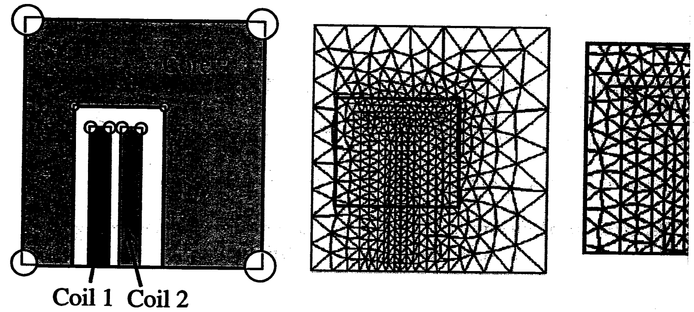
a) analysis model b) generated mesh c) zoom-up view

Fig. 7 Mesh generation for model of a coil in the air.

$$\begin{aligned} r(1,0,0) &= r_1 = A \\ r(0,1,0) &= r_2 = A\ e^B \\ r(0,0,1) &= r_3 = A\ e^C \end{aligned} \quad . \tag{10}$$

4. Applications

In order to verify the usefulness of the proposed method, it was applied for automatic mesh generation of several test models. In this paper we present only two simple models: model of a coil in the air (Fig. 7) and model of a transformer's window area (Fig. 8). From Figs. 7 and 8 it is apparent that the proposed method results with meshes with graded mesh density according to the problem description and user's desired mesh topology.



a) analysis model b) generated mesh c) zoo

Fig. 8 Mesh generation for model of transformer window are

5. Conclusion

A new dynamic model for automatic mesh generation using bubble sy sented. It uses a frontal method for generation of a initial set of nodes us based bubble system that obeys the Second Newton's Law of Dynamics. element mesh is generated using the obtained set of nodes according to t triangulation method. The proposed method requires small amount of i generates finite element meshes with graded mesh density according to th with less computation effort and cost even for domains and structures wit ometry. Although, in this paper, only 2-D application of the proposed m it can be easily extended into 3-D space.

References

[1] H. Yamashita and E. Nakamae, An automatic triangular mesh gener element analysis, *Transaction of Information Processing Society of J* No. 3, (1980) (in Japanese).

[2] T. Taniguchi, Automatic mesh generation for FEM, *Morikita Pre* Japanese).

[3] K. Shimada, Physically-based automatic mesh generation, *Journal o Society for Simulation Technology*, Vol. 12, No. 1, (1993) (in Japanese