

|      |
|------|
| 学術論文 |
|------|

# バブルシステムを用いた要素自動分割法

Automatic element subdivision using Bubble system

|       |         |                 |        |
|-------|---------|-----------------|--------|
| 村川 了  | (広島大学院) | Čingoski Vlatko | (広島大学) |
| 金田 和文 | (広島大学)  | 山下 英生           | (広島大学) |

Ryo MURAKAWA, Student member of JSAEM, Graduate school of Engineering,  
Hiroshima University  
Čingoski VLATKO, Hiroshima University  
Kazufumi KANEDA, Hiroshima University  
Hideo YAMASHITA, Member of JSAEM, Hiroshima University

A new method for automatic mesh generation using dynamic bubble system is presented in this paper. The proposed method has two separate processings: one which generates nodes inside the analysis domain using physically-based system of bubbles, and the second one for automatic generation of finite elements according to the Delaunay algorithm using 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 for the entire region. In the proposed method the density of the mesh easily can be controlled using simple exponential functions which allows obtaining desired mesh density with modest amount of input data for a very short computation time. The proposed meshing method is applicable for automatically meshing geometrically complicated shapes and structures such as those found in various electromagnetic devices like rotating machines and transformers.

*Key Words* : Bubble system, Delaunay algorithm, Automatic element subdivision, van der Waals forces

## 1 緒言

計算機の性能向上に伴い、有限要素法が様々な分野の大規模なシミュレーションに用いられるようになってきた。一方、解析モデルの巨大化や複雑化は、要素分割にかかる時間を著しく増大させている。特に、要素形状や要素分布などを考慮に入れて分割を行う場合、これを手動で行うと、これに費やす労力は、解析時間の数倍から数十倍になると言われている。そのため、要素分割の完全な自動化に関する研究が数多く報告されている。

要素自動分割の必要条件は、任意形状の領域が分割可能であること、小さい要素と大きい要素を適宜配置できること、要素歪みを排除できること、入力データが少なく簡便であること、対話的環境で要素分割が行えることなどである。

これまでに、この条件を満足させようと試みた様々な要素自動分割法が報告されており、それらを分類す

ると、部分領域分割法 [1][2]、4分木や8分木に代表される階層的空間分割法 [3][4][5]、再帰的二分割法 [6]、ノード結合法 [7][8] の4つに分類される。しかし、これらの方法にもまだ解決されていない次のような問題点がある。部分領域分割法は、入力データが多量、階層的空間分割法と再帰的二分割法は要素形状の制御が困難、ノード結合法は、解析者の要求する要素分布を満たすように節点を解析領域内に配置することである。近年の要素自動分割の傾向として、考え方の単純さ、二次元・三次元領域の統一的な扱い、Delaunayの方法 [9] の優秀さなどの理由により、ノード結合法に注目が集まっている。

ノード結合法の問題点を解決する一手法として、バブルの物理モデルを用いた手法が報告されている [10]。しかし、文献 [10] には、解析領域内のバブルの半径決定方法や初期配置について明確に記述されていない。ところで、この方法では、バブルの初期配置によって計算時間が大きく影響を受け、場合によっては解が収

束しない場合が生ずる。そこで、我々は、解析領域内のバブルの半径決定法や初期配置について有効な方法を提案し、2次元磁界問題に適用し、良い結果が得られたので報告する。

## 2 バブルシステムについて

バブルは、半径と質量を持つ球状粒子である。バブルシステムは、複数のバブルが次のようにニュートン力学に基づいた運動方程式によって支配される物理モデルである。

$$m_i \frac{d^2 x_{ij}}{dt^2} + c_i \frac{dx_{ij}}{dt} = f_{ij} \quad (j = x, y, z) \quad (1)$$

ここで、 $m_i, c_i$ は、それぞれバブル  $i$ の質量と粘性抵抗であり、 $x_{ij}$ はバブル  $i$ の中心の  $j$ 成分の座標、 $f_{ij}$ はバブル  $i$ に働く  $j$ 方向の外力である。

この運動方程式を解くことにより、バブルの移動の様子をシミュレーションできる。外力は、ファンデルワールス力をモデルとして、図1の様に互いのバブルが接した状態を平衡状態とし、互いに近づきあうと斥力が働き、遠ざかると引力が働くものとする。また、

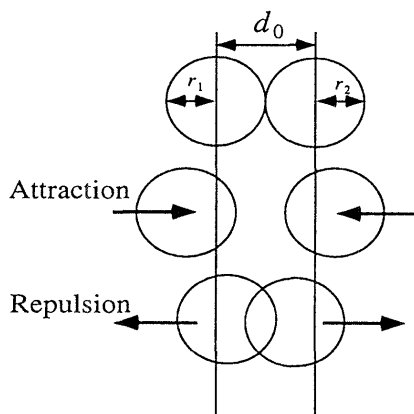


Fig. 1: Type of forces

ファンデルワールス力は、2つのバブルの中心間の距離  $d$ を変数として次式のように書くことができる [11].

$$f(d) = 12d_0^6 \epsilon \left( \frac{d_0^6}{d^{13}} - \frac{1}{d^7} \right) \quad (2)$$

ここで、 $d_0$ と $\epsilon$ は、それぞれ2つのバブルが接した状態のバブルの中心間の距離(安定距離)と2つのバブルの結合エネルギーを示す。実際にシミュレーション

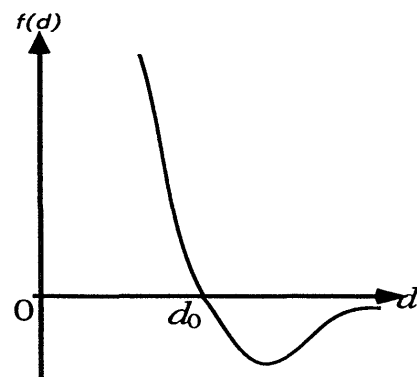


Fig. 2: van der Waals forces

を行うときは、式(2)をそのまま用いると図2に示すように、2つのバブルが重なり合ったときに力が無限大になったり、遠く離れたバブル間の無視できるぐらい小さい力を考慮しなければならないので、次式のように、式(2)を3次式で近似することにする。

$$f(d) = Ad^3 + Bd^2 + Cd + D \quad (3)$$

なお、未知数  $A, B, C, D$ は、次の条件より決定することができる。

$$\begin{aligned} f(d_0) = 0, \quad f(d_1) = 0, \\ f'(0) = 0, \quad f'(d_0) = -k_0 \end{aligned} \quad (4)$$

ここで、 $d_1 (> d_0)$ は、バブルの力が及ぶ範囲で  $d_1 = 1.3d_0$ とし、 $k_0$ は  $d_0$ におけるファンデルワールス力の微分係数の絶対値で次式のように表すことができる。

$$k_0 = \frac{72\epsilon}{d_0^2} \quad (5)$$

なお、 $k_0$ の値は、バブルの移動シミュレーションを行うバブル群ごとに一定の値を設定する。この決め方については3.1節で述べる。 $k_0$ の値を一定にすることによって、初期配置において隣接するバブル間の距離が、 $d_0$ に近い状態に配置されたとき、容易にブロック内のバブルに作用する力の釣り合いをとることができる。また、 $d_1$ を設定することによって、考慮しなければならない力の範囲を限定できるので、計算時間を短縮することができる。なお、結合エネルギー $\epsilon$ は、1とする。また、粘性抵抗の計算法は、文献[10]の方法を用いて以下のように設定している。

$$\text{辺バブル} : c_i = 2\sqrt{m_i k_0} \quad (6)$$

$$\text{面バブル} : c_i = \sqrt{2(2 + \sqrt{3})}m_i k_0 \quad (7)$$

$$\text{質量} : m_i = \pi r_i^2 \quad (8)$$

### 3 バブルシステムを用いた要素分割法

#### 3.1 提案手法の概要

本手法は、図 3(a) に示すようにバブルシステムを用いた節点発生部と Delaunay の方法を用いた三角形発生部とからなる。

節点発生部の入力データは、解析領域の形状データと各頂点に配置するバブルの半径である。また、各バブルは、図 3 に示すように制約条件の多いものから順に配置される。すなわち、頂点バブル、辺バブル、面バブルの順である。ここで、頂点バブルは、頂点上に配置されたバブルのことを言い、入力データにより指定された半径のバブルである。辺バブルは、その移動が辺上に制限されたバブルのことを言い、面バブルは、面上を自由に移動できるバブルのことを言う。

この順に配置されたバブルが力の平衡状態になるまで、物理シミュレーションを行い、最終的に得られたバブルの中心の座標を元に Delaunay の方法を用いて三角形分割する。

バブルシステムを用いた要素自動分割では、次の有用な特徴が得られる。

- 入力データが少ない
- 要素分布が連続的に制御可能
- 最終的に得られる要素形状の歪みが少ない

まず、これらの点を実現するために、ユーザは解析領域形状と、そこに存在する物質形状および形状を決定する各頂点付近の要素の大きさに依存するバブルの半径のみを入力する。

続いて、解析領域中の頂点だけを用いて、Delaunay の方法によって解析領域全体を三角形ブロックに分割する。得られた三角形ブロックを用いて、凸四角形ブロックを作成する。このとき、凸四角形ブロックが作成できないものは、三角形ブロックのまま残しておく。

次に各ブロックを形成する辺上のバブルを、その辺の端点に与えた頂点バブルの半径を用いて発生させ、バブルの移動を物理シミュレーションにより行う。

最後に、四角形ブロック又は三角形ブロック内に面バブルを発生させた後、解析領域内の全バブルの物理シミュレーションを行う。

一般に、電磁界問題を有限要素法を用いて解析する場合、未知変数は磁気ベクトルポテンシャル又は電位である。開領域の問題に対しては、これらの分布は、値の高いところから低いところに指数関数的に変化することが知られている。そこで、提案手法では、辺バブル、面バブルの半径を頂点バブルの半径から指数関数的に変化するようにした。なお、式 (5) の  $d_0$  の値は、辺バブル、面バブルのそれぞれに対して次のように決める。辺上においては、 $d_0$  を始点と終点の頂点バブルの半径の和とし、面上においては、ブロックを形成する頂点バブルの半径の平均値の 2 倍の値とする。

#### 3.2 辺バブルの発生方法

本節では、辺バブルの移動シミュレーション時間を短くするために、辺バブルの初期位置とその半径の決定方法について述べる。これらを決定するために、辺の 2 つの端点のバブルの半径  $r_1$  と  $r_2$  および辺の長さ  $l$  から決まる半径曲線 (式 (9)) とこの曲線の始点と終点を通る直線 (式 (10)) を利用する。但し、 $r_1 < r_2$  とする。ここで、半径曲線は、前節で述べた辺上の各バブル半径の大きさが指数関数的に分布するように、各

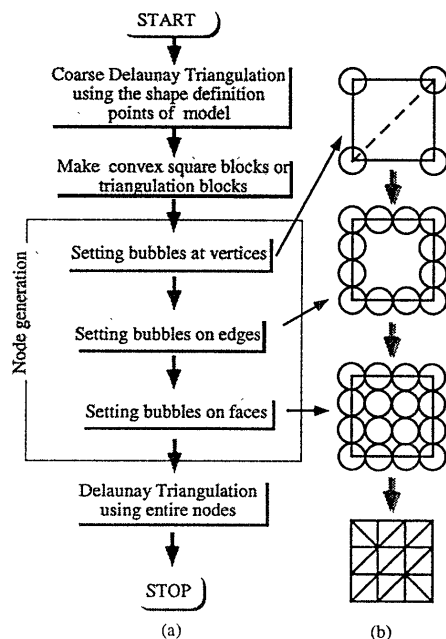


Fig. 3: Flowchart

バブルの半径を決める曲線であり、次式で表わすことができる。

$$r(x) = Ae^{Bx} \quad (9)$$

ここで、

$$A = r_1$$

$$B = \frac{\log(r_2/r_1)}{\ell}$$

また、この曲線の始点と終点を通る直線は、

$$y = ax + b \quad (10)$$

ここで、

$$a = \frac{r_2 - r_1}{\ell}$$

$$b = r_1$$

で表すことができる。

辺バブルの発生方法および初期配置は以下の手順で行う。すなわち、辺の両端点から辺中央に向けて順次辺上にバブルを配置する。ただし、頂点バブルの半径の小さいほうから行う。

**手順 1** まず、半径  $r_1$  の頂点バブルの方から、辺バブルの初期配置を求めるために  $x_0=r_1$  として **手順 2** へ (図 4(a) 参照)。

**手順 2** 辺バブルの中心座標  $x$  を次式により求める。

$$x = \frac{r_1 + x_0}{1 - a}$$

ここで、 $x > \ell/2$  ならば、**手順 4** へ。そうでなければ **手順 3** へ。

**手順 3** 座標  $x$  におけるバブルの半径  $r$  を次式により求める。

$$r = Ae^{Bx}$$

そして、

$$x_0 = x + r$$

として、**手順 2** へ。

**手順 4** 次に半径  $r_2$  のバブルの方から始める。ここで、 $L=\ell$ ,  $r_0=r_2$  として次へ。

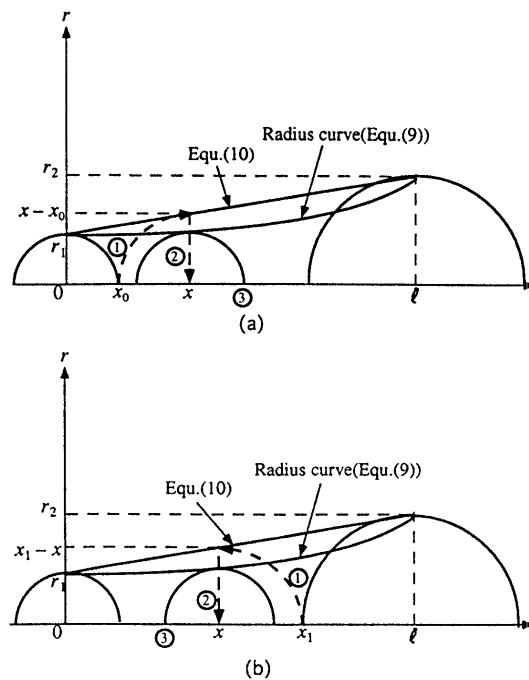


Fig. 4: Initial position of edge bubbles

**手順 5** 図 4(b) に示すように座標  $x$  の点を次式を用いて求める。

$$x_1 = L - r_0$$

$$x = \frac{x_1 - b}{a + 1}$$

**手順 6**  $x < \ell/2$  ならば、処理を止める。

**手順 7** 座標  $x$  におけるバブルの半径  $r$  を次式により求める。

$$r = Ae^{Bx}$$

そして、

$$r = r_0, L = x$$

として、**手順 5** へ。

この手順で、辺バブルを発生させることにより、辺上に大きさが連続的に変わるバブルを配置することができ、初期配置の状態ではバブルに及ぼす外力が 0 近くになっているので、物理シミュレーションにかかる時間を大幅に減らすことができる。

### 3.3 面バブルの初期配置および半径の決定方法

面バブルの発生方法は、概要で述べたように、解析モデルをすでに凸四角形か三角形の領域に分割しているので、それぞれの領域に対して以下のように行う。

● 四角形の場合

手順 1 凸四角形を長さが 1 の正方形にマッピングする (図 5 参照). この時, 凸四角形の頂点バブルの半径が最小のものを (0,0) にマッピングする.

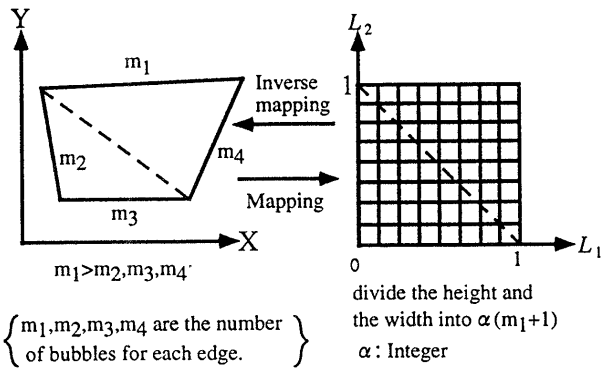


Fig. 5: Mapping from convex square to unit square

手順 2 四角形の各辺の辺バブルの個数の中から最大のものを見つけ  $n_{max}$  とする. 例えば, 図 5 では,  $m_1$  が最大となる.

手順 3 マッピングした単位正方形の縦と横を各々  $\alpha (n_{max} + 1)$  ( $\alpha$ : 整数) 分割し, 正方形内の交点を, 面バブルを置く候補点とする.

手順 4 この候補点を 1 つずつ順に逆マッピングして, 既に配置済みの辺バブルおよび既に配置された面バブルとの間に次の条件を満足すれば, その点にバブルを置く (図 6 参照).

$$\beta \cdot (r_{can} + r_{old}) < d$$

$r_{can}$  : 候補点のバブルの半径

(半径の決定方法は, 3.4 節で述べる)

$r_{old}$  : 既存のバブルの半径

$d$  : バブル間の中心距離

$$0.5 < \beta \leq 1.0$$

これを候補点が無くなるまで繰り返す. ここで,  $\alpha$  は, ブロック内に発生させる候補点の数を調整するために用い,  $\beta$  は, 初期配置でバブルの重なり具合を調整するために

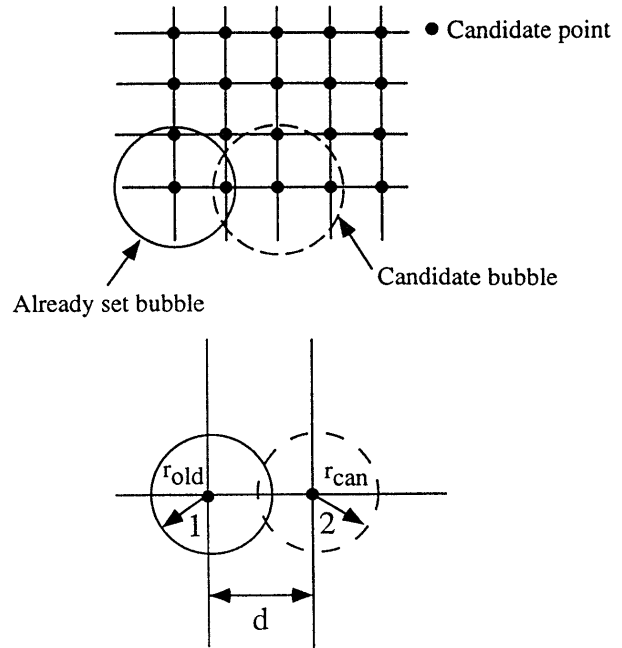


Fig. 6: Set bubble on face

用いる. すなわち,  $\alpha$  を大きくすると, 探索する候補点の数が増え,  $\beta$  を大きくすると, 隣り合うバブルがほぼ接した状態で配置されることになる.

手順 4 で, 候補点を 1 つずつ選ぶ順序は, 図 7(a) に示すように単位正方形を領域 A と B に分割し, まず領域 A において, 原点 (0,0) に近い候補点から,  $L_1$  および  $L_2$  方向に交互に領域 A のすべての候補点について探索を行う. 次に領域 B において点 (1,1) に近い候補点から同じように全ての候補点について探索を行う. このように行うことによって, 対角線に対して比較的対称な要素分割を得ることができる.

● 三角形の場合

手順 1 四角形の場合と同様に一辺の長さが 1 の直角二等辺三角形にマッピングし, 三角形の各辺の辺バブルの最大個数  $n_{max}$  を見つけ, 正方形の縦と横を  $\alpha(n_{max} + 1)$  分割する. ここで, 候補点としては,  $L_1 + L_2 < 1$  の条件を満たさねばならない.

手順 2 あとは, 四角形の場合と同様な処理を行う.

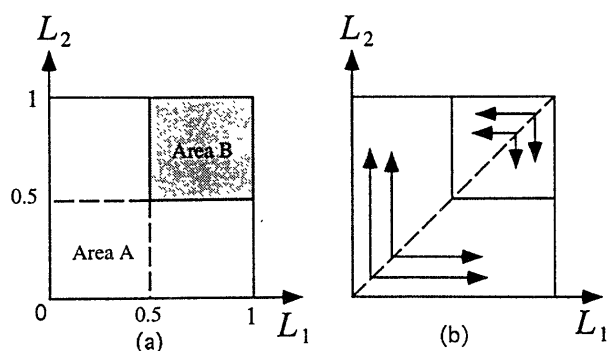


Fig. 7: Search for candidate points

このように候補点の探索を行うことによって、ほぼ過不足無く解析領域中にバブルを充填することができ、辺バブルのときと同様な理由により物理シミュレーションにかかる時間を短縮することができる。

### 3.4 面バブルの半径決定方法

解析領域内の面バブルの半径決定方法は、三角形ブロックを用いて行う。すなわち、三角形ブロックを形成する頂点バブルの半径を用いて次式で計算する。

$$r(L_1, L_2, L_3) = Ae^{BL_2 + CL_3} \quad (11)$$

ここで、 $L_1, L_2, L_3$ は、面積座標であり、 $A, B, C$ は、三角形ブロックの各頂点バブルの半径の $r_1, r_2, r_3$ から、次式で求める。

$$\begin{aligned} r(1, 0, 0) &= r_1 = A \\ r(0, 1, 0) &= r_2 = Ae^B \\ r(0, 0, 1) &= r_3 = Ae^C \end{aligned}$$

## 4 $\alpha$ と $\beta$ に対する要素形状の検討

面バブルの設定のときに用いるパラメータ $\alpha$ と $\beta$ の値が最終的に生成される三角形要素の歪みにどのように影響するか調べた。面バブルは、四角形領域か三角形領域に配置されるので、検討モデルとして、図8に示すような正方形の解析領域を持つモデルを用いた。ここで、頂点 $A, B, C, D$ のバブルの半径を、以下のような4つのケースについて実験を行った。

**ケース 1:** 頂点 $A, B, C, D$ のバブルの半径をすべて等しくした場合(半径:40mm)

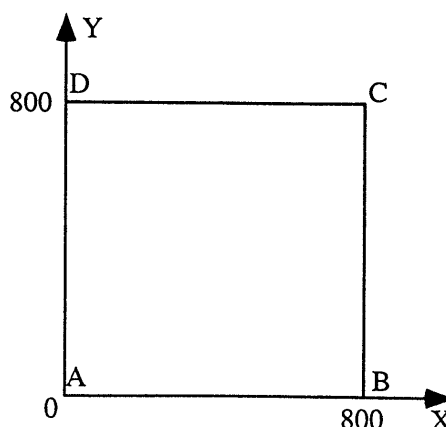


Fig. 8: Test model

**ケース 2:** 頂点 $A, B, C, D$ のバブルの半径をそれぞれ40,80,100,80(mm)とした場合(4頂点の半径の比がそれほど大きくない場合)

**ケース 3:** 頂点 $A, B, C, D$ のバブルの半径をそれぞれ10,80,100,80(mm)とした場合(4頂点の半径の比が大きい場合)

**ケース 4:** 頂点 $A, B, C, D$ のバブルの半径をそれぞれ10,30,100,70(mm)とした場合(4頂点の半径が全て異なる場合)

ここで、要素の歪みを調べるために次式の $Q$ を用いた。

$$Q = \frac{r_{out}}{2r_{in}} \quad (12)$$

$r_{out}$  : 三角形要素の外接円の半径

$r_{in}$  : 三角形要素の内接円の半径

この式は、要素が正三角形のとき、最小値1をとり、要素が歪むにつれて、値が大きくなる。

それぞれのテストケースの $Q$ の最大値と平均値の結果を図9~12に示す。 $Q$ の平均値について考えてみると $\alpha$ の値を大きくすると、 $Q$ は1に近づく傾向にある。

一方、 $Q$ の最大値について考えてみると、 $\alpha$ の値を増加させてもかならずしも小さくなる、すなわち要素形状が良くなるとは限らない。しかし、これは、要素形状修正法(ラプラシアン法や辺スワッピング)を要素生成時に適用することによって、 $Q_{max}$ を小さくすることができ要素形状を良くすることが可能であると思われる。結論として、 $\alpha=3, \beta=0.75$ にとるのが良いと思われる。また、 $\alpha=3, \beta=0.75$ の時の4つのケースの分割結果を図13に示す。

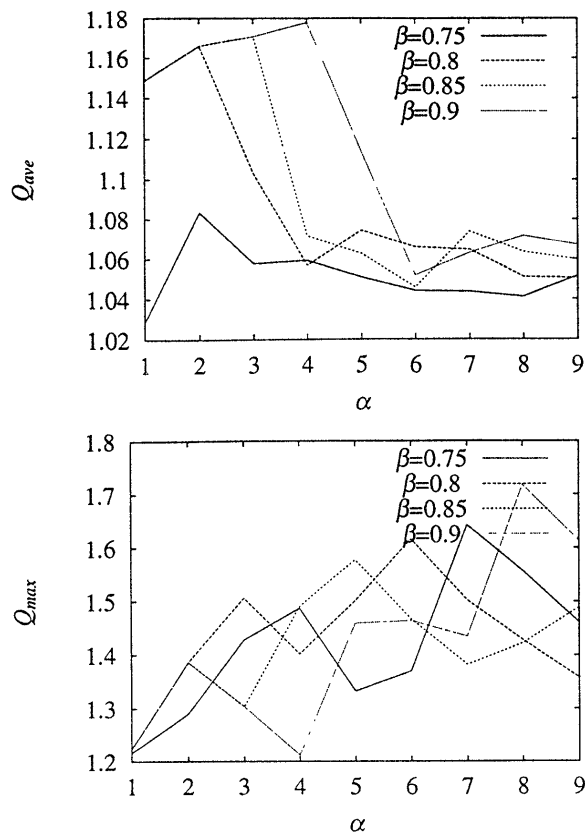


Fig. 9: The maximum value  $Q_{max}$  and average value  $Q_{ave}$  of mesh quality for case1

## 5 適用例

提案手法を導体の磁場解析モデル、単相変圧器のモデル、計器用変成器のモデルに適用し、有用性を示す。このとき全てのモデルで、 $\alpha=3, \beta=0.75$ とした。

最初に、導体周囲の磁場解析モデルの結果を示す。図 14 (a) に解析モデル、(b) に入力データとしての頂点バブル、(c) に分割結果、(d) に導体付近の拡大図を示す。このモデルは、開領域問題であり、導体の頂点に与えたバブルの半径と解析領域の導体から離れた頂点に与えたバブルの半径の大きさは、40 倍の差がある例であるが、導体付近から外側に向けて要素の大きさは、しだいに大きくなり良い結果が得られている。

次に、単相変圧器の結果を示す。図 15(a) に解析モデル、(b) に入力データとしての頂点バブル、(c) に分割結果、(d) に磁束が集中する部分の拡大図を示す。このモデルは、閉領域問題で複数の物質があるモデルで、各頂点のバブルの比が、最大で 5 であるような例である。要素の大きさは、1 次および 2 次コイル付近

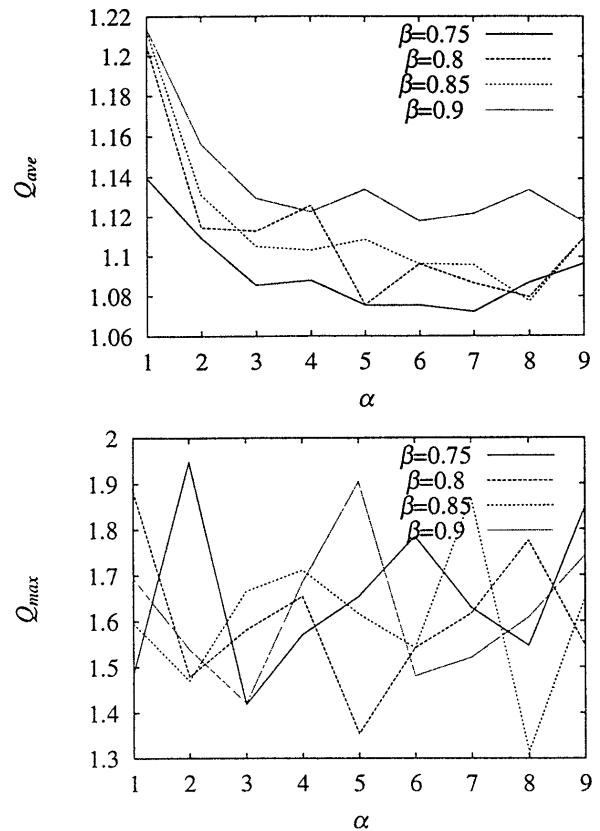


Fig. 10: The maximum value  $Q_{max}$  and average value  $Q_{ave}$  of mesh quality for case2

から鉄心方向に徐々に小さくなっているのが分かる。

最後に、計器用変成器の分割結果を示す。図 16(a) に解析モデルと入力データとしての頂点バブル、(b) に分割結果を示す。このモデルは、複雑な形状をした閉領域モデルで、頂点バブルの大きさの比は 20 倍である例であるが、頂点バブルの小さいところから大きいところに向けて要素の大きさは、しだいに大きくなり良い結果が得られている。各モデルの分割結果の  $Q_{max}$  と  $Q_{ave}$  および計算時間を表 1 に示す。但し、使用計算機は、Silicongraphics 社の Power Indigo2 である。

## 6 結論

バブルシステムを用いた有限要素自動分割に対する一手法を提案した。本手法は、入力データが少なく、少ない計算時間で歪みの少ない要素が得られるのが特徴である。

将来の研究としては、アダプティブ法への適用であ

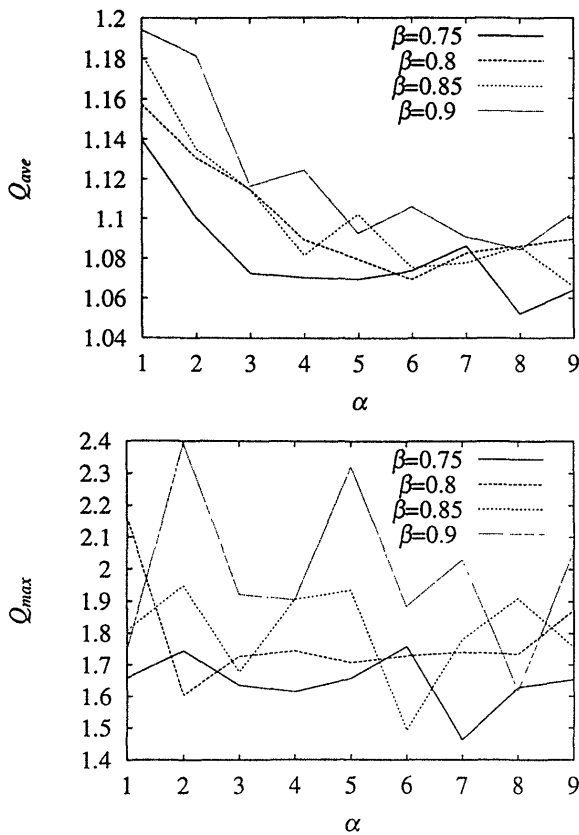


Fig. 11: The maximum value  $Q_{max}$  and average value  $Q_{ave}$  of mesh quality for case3

り、誤差評価の結果により頂点バブルの半径を導出し、このシステムを用いて、要素の自動生成を行う方法の開発、および本手法を三次元に拡張した三次元自動要素分割があげられる。

(96年11月15日受付)

#### 参考文献

- [1] E. A. Sadek, A scheme for automatic generation of triangular finite elements, International Journal for Numerical Methods in Engineering, Vol.15, (1980), pp.1813-1822
- [2] B. Joe, R. B. Simpson, Triangular meshes for regions of complicated shape, International Journal for Numerical Methods in Engineering, Vol.23, (1986), pp.751-778
- [3] M. A. Yerry, M. S. Shephard, A modified

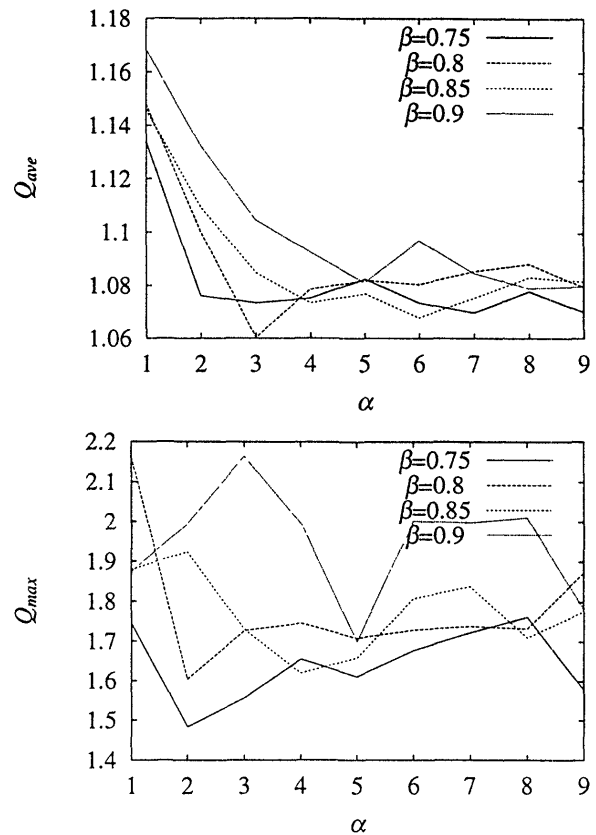
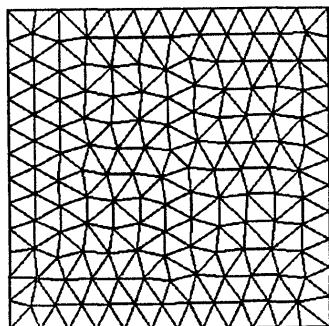


Fig. 12: The maximum value  $Q_{max}$  and average value  $Q_{ave}$  of mesh quality for case4

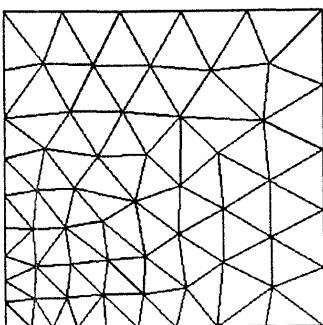
quadtree approach to finite element mesh generation, IEEE Computer Graphics and Applications, Vol.3(1), (1983), pp.39-46

- [4] R. Perucchio, M. Saxena, A. Kela, Automatic mesh generation from solid models based on recursive spatial decompositions, International Journal for Numerical Methods in Engineering, Vol.28, (1989), pp.2469-2501
- [5] P. L. Baehmann, S. L. Wittchen, M. S. Shephard, K. R. Grice, M. A. Yarry, Robust, geometrically based, automatic two dimensional mesh generation, International Journal for Numerical Methods in Engineering, Vol. 24, (1987), pp.1043-1078
- [6] B. Wordenweber, Finite element mesh generation, Computer-Aided Design, 16(5),

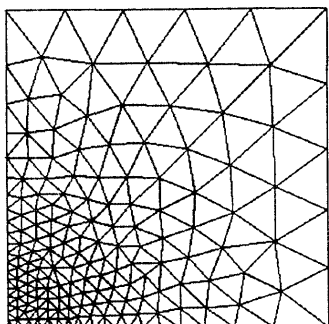




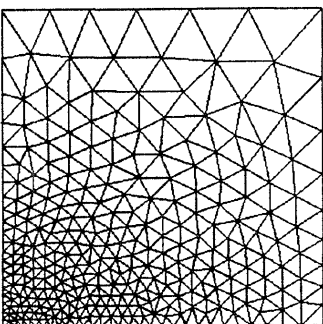
case1



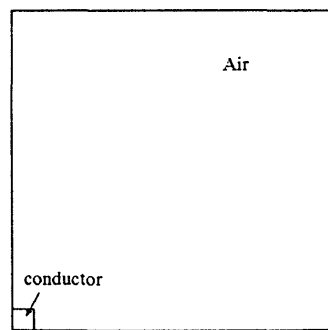
case2



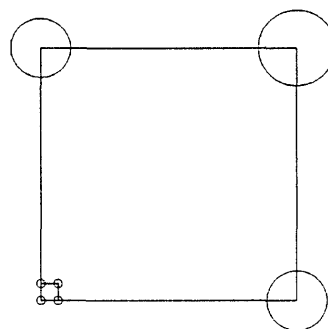
case3



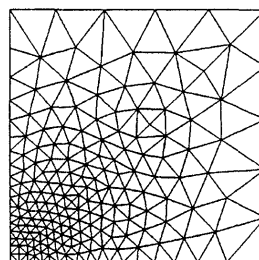
case4



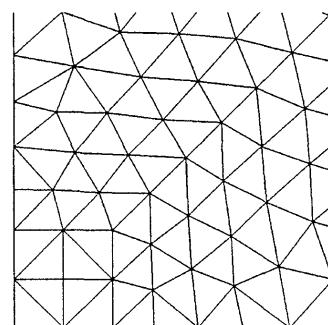
(a) Analysis model



(b) Input data



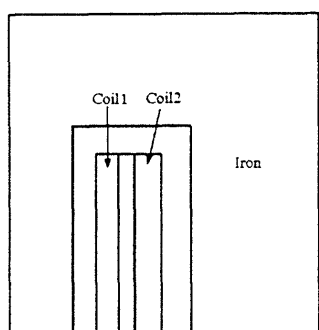
(c) Division map



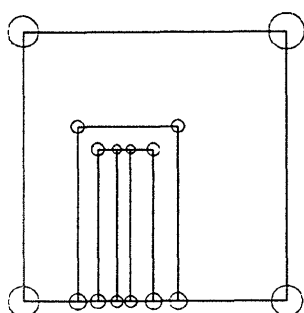
(d) Zoom in around conductor area

Fig. 13: Division maps of case1 ~ 4

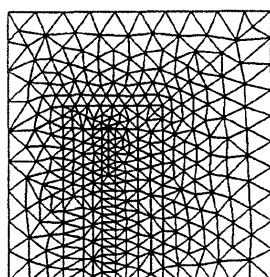
Fig. 14: A model of conductor in the air



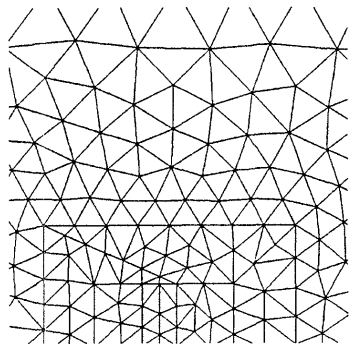
(a) Analysis model



(b) Input data

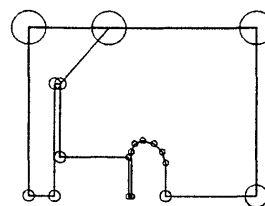


(c) Division map

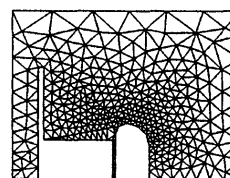


(d) Zoom in around coil area

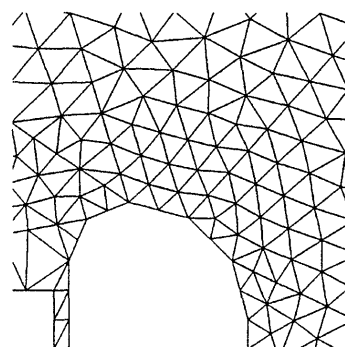
Fig. 15: Transformer model



(a) Input data



(b) Division map



(c) Zoom in

Fig. 16: Potential transformer model

Table. 1:  $Q_{max}$ ,  $Q_{ave}$  and calculation time

|                      | A model of conductor<br>in the air | Transformer<br>model | Potential<br>transformer model |
|----------------------|------------------------------------|----------------------|--------------------------------|
| number of nodes      | 193                                | 349                  | 368                            |
| number of elements   | 338                                | 652                  | 639                            |
| $Q_{max}$            | 1.5199                             | 2.1559               | 3.0207                         |
| $Q_{ave}$            | 1.0805                             | 1.1215               | 1.1083                         |
| calculation time (s) | 45                                 | 15                   | 13                             |

(1984), pp.285-291

- [7] J. C. Cavendish, Automatic triangulation of arbitrary planar domains for the finite element meshes, *International Journal for Numerical Methods in Engineering*, Vol.8, (1974), pp.679-696
- [8] 矢川 元基, 吉村 忍, 中尾 和弘, 鶴 大悟, あいまい知識処理手法と計算幾何学に基づく大規模自動要素分割 (二次元, 三次元ソリッドおよび三次元シェルへの応用), *日本機械学会論文集 A*, (1992), pp.1245-1253
- [9] N. P. Weatherill, A METHOD FOR GENERATION IRREGULAR COMPUTATIONAL GRID IN MULTIPLY CONNECTED PLANAR DOMAINS, *International Journal for Numerical Methods in Engineering*, Vol.8, pp.181-197, 1988
- [10] 嶋田 憲司, 物理モデルによる自動メッシュ分割, *日本シミュレーション学会誌*, Vol.12(1), (1993), pp.11-20
- [11] 日本機械学会編, 原子・分子モデルを用いる数値シミュレーション, (1996), コロナ社