

三次元凸体の四面体有限要素自動分割

谷口健男*・太田 親**

三次元体を対象とした有限要素解析では対象の要素分割が必須となる。本研究は三次元体の内形状が凸な系に対して有効な要素自動分割法を提案したものである。本研究ではまず、提案した方法が厳密にデラウニー三角分割となることを証明し、ついで、三次元におけるビン・ソート法や、点を包含する四面体の新たな探査法を提案して四面体要素生成アルゴリズムの高速化を図っている。

Keywords : Delaunay triangulation, 3D convex body, tetrahedron, bin sorting

1. はじめに

近年、電子計算機の発達により三次元体の有限要素解析が行われるようになり、その解析の前処理にあたる要素自動分割法の開発が緊急の課題となっている。たとえば、三次元体中での亀裂伝播解析を行おうとすれば、亀裂の進展に対応して要素分割を行わなければならない。しかしながら、三次元体を対象とした場合、全体形状の複雑さ、亀裂破面の幾何学的複雑さ等の理由により要素分割データを解析者が手動で作成することは、ほとんど不可能となる。たとえそれが可能であっても三次元体の内部の要素配列が適切であるか、また確実に分割されているかを確認するには多大のコストが要求される。したがって、対象系が与えられたとき、それを正四面体に近い要素の集合に分割する方法、およびその自動化手法の開発は三次元解析では不可欠である。

一般的に三次元体の要素分割法の基本的な要求事項は

- #1 完全自動分割法：コスト低減に不可欠。
- #2 要素分割が隙間なく、重なりなくできるような確実な分割法。
- #3 どのような節点配置に対しても分割が可能であること。
- #4 可能な限り正四面体に近い要素分割が得られること。
- #5 迅速な手法であること。

に集約されるであろう。

これらの諸要求に応える基本的なものとして今日、八分木法とデラウニー分割法がある。前者は空間内にまず粗く格子状に節点を分配させ、ついで系の境界形状を十分に表現できるまで境界周辺部に節点を密に配置し、最

終的にメッシュ分割を完成させる方法である^{1)~3)}。

一方後者は節点配置が基本条件であり、それらの節点を用いて作られた四面体要素が最も正四面体に近くなるように要素分割を行う手法である。両者を比較すると、前者はデータ構造が簡単であることよりそのプログラム化が容易である反面、境界認識を行わねばならないこと・作られた要素の幾何学的な形状に問題がでる危険性をもっていること等の欠点を有している。後者は出来上がった要素が、与えられた節点配置に対して最も正四面体に近いという保証があることより、有限要素モデル作成道具として適している。なお、文献2), 3)においては、 n 次元空間を対象とした一般的な分割法が示されている。

デラウニー分割法は数学的にはポロノイ分割の相対問題となっている。このポロノイ分割は、三次元空間内に任意に分布された節点群を対象として、節点数に等しい凸多面体の集合に分割する方法である。多面体内にはただ1個の節点しか含まれず、また多面体表面を構成する凸多角形は隣接する2個の節点の対称面となっている。この性質によりその相対問題であるデラウニー分割で得られる四面体はその外接球の内部に決して他の節点を含まないことになる。このポロノイ分割あるいはデラウニー分割は航空機まわりの三次元流体解析の分野等ですで行われている [たとえば文献4)]。

一方、二次元空間のデラウニー分割に関しては、Sloan が三角形 (スーパートライアングル) を導入し、その内部に順次節点を発生させ、直接三角形を作成するという手法を提案している⁵⁾。このスーパートライアングルの導入はアルゴリズムの単純化・高速化に大きな役割を果たしている。しかしながら、このSloanの考えは、三次元空間の四面体分割にはいまだ取り入れられていない。

本研究では、Sloanの提案したスーパートライアングル

* 正会員 工博 岡山大学助教授 工学部共通講座
(〒700 岡山市津島中 3-1-1)

** 学生会員 岡山大学大学院生

ルの概念を拡張してすべての節点を包括するスーパーテトラヘドロンを導入し、それをを用いた三次元空間の四面体分割法の提案を行う。この新たな四面体分割法を提案するに際して必要な数学的証明をまず行い、次いで現実のプログラム化、およびその高速化に必要な道具の説明を行い、最後に分割例とそれに要する演算時間の検討を行う。なお、ここで示す要素分割法はSloanの考え方とは異なり、まず空間のポロノイ分割を行い、それを四面体分割に置き換えるというものである。

2. 三次元凸体の四面体分割の基礎理論

三次元空間内に設置された節点をを用いたデラウニー分割の意味での四面体分割を行うとき、空間内に節点を順次設置し、その都度デラウニー三角分割を行うことを考える。この手法を採用すると、新たに1点を設置したときそれによりすでに完成している四面体集合のどの部分を修正しなければならないのか、そして、それをどのように四面体分割すれば新たなデラウニー分割となるかを示す必要があり、本節ではこの点について考察を加える。なお、文献4)には以下の4つの結果のうち、第3, 4に関して、違った表現ではあるが定理として示されている。

いま、三次元空間内に設定された n 個の節点をを用いて四面体分割ができていると仮定する。この段階で新たに1個の点をその空間内に追加したときのデラウニー分割を求めることを考える。以下の文中においては次に示す記号を用いる。

- Tet(ABCD) : 節点 A, B, C, D を頂点とする四面体
- Sph(ABCD) : 四面体 Tet(A, B, C, D) の外接球
- TRi(ABC) : 三角形 A, B, C

「四面体分割された三次元空間内に $n+1$ 番目の点 P を新たに設定したとき、その点を包含する外接球をもつ四面体は互いに面接合する。」

証明

いま、追加した点 P が Tet(ABCD) 内に位置すると仮定すると、その外接球 Sph(ABCD) は明らかにその点を包含する。さらに、 P が Tet(ABCD) に隣接しない他の四面体の外接球 Sph(IJKL) にも包含されているとしよう。Sph(IJKL) は Tet(ABCD) と Tet(IJKL) の間に位置するすべての四面体を通り、しかも他の節点をその外接球内部に包含することなく節点 P に到達していることになる。

いま、Sph(IJKL) に貫かれる四面体の一つを Tet(EFGH) で表すと図-1 に示すように

$$\text{Sph(IJKL)} \cap \text{Sph(EFGH)} \neq \emptyset \dots\dots\dots (1)$$

Tet(ABCD), Tet(EFGH), Tet(IJKL) の位置関係より、 P は $\text{Sph(EFGH)} \cap \text{Sph(IJKL)}$ の中に位置し、したがって P は Sph(EFGH) に包含される。

上に示した Sph(EFGH) と Sph(IJKL) の関係は Tet

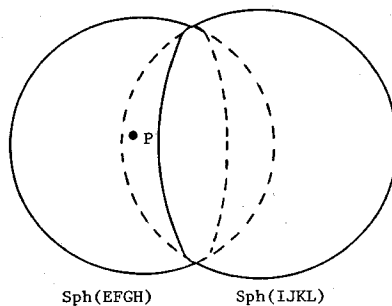


図-1 2個の外接球と点Pの関係

(ABCD) と Tet(IJKL) の間に位置するすべての四面体に対して成り立ち、したがって P はそれらすべての外接球内に含まれる。よって、それらは互いに面接合する。「上で得られた四面体の集合の共通面(三角形)を取り除くことにより多面体を作ることができる。」

証明

上に示したように P を包含する四面体は互いに面接合している。したがって互いに隣接する四面体は三角形を共有していることより、その共有面である三角形を取り除けば表面を三角形の集合で覆われた多面体ができる。

「この多面体表面のそれぞれの三角形と P を結ぶことにより、多面体を四面体の集合で満たすことができる。」

証明

この証明は多面体上の任意三角形と P により四面体を作成できることを示せば良い。いま、多面体上の任意の三角形を取り出すと、本来はこの三角形と多面体上の他の一点とで四面体を構成している。さらに、この四面体は外接球上に位置し、 P はその内部に包含されることは明らかである。したがって、 P と三角形の頂点は直線で必ず結び付けることができ、四面体を構成できる。多面体上のすべての三角形に関して同様にいえることより、多面体は四面体の集合で必ず埋めることができる。「上で得られた四面体集合は $(n+1)$ 点におけるデラウニー分割である。」

証明

空間を P を包含する外接球をもつ四面体の集合で作成できる多面体内部とその外部に分離して証明を行う。内部に関して P を用いずに四面体を作成しようとすればその頂点はすべて多面体上の節点を用いることになる。このようにしてできる四面体は必ず他の節点(たとえば点 P) をその内部に包含する。したがって、上で述べた多面体のデラウニー分割で得られる四面体はすべて P をその頂点として含まなければならない。

次に、多面体の外部について考える。上で示した多面体の内部に新たに作成された四面体とそれに接する外部

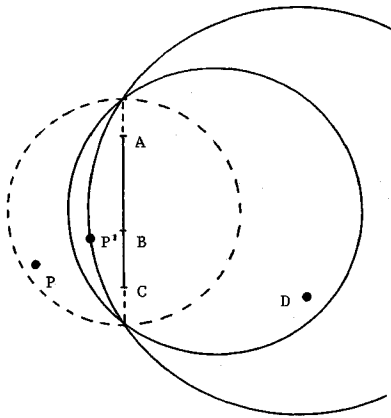


図-2 2個の外接球の関係

に四面体を考える。いまこれらを Tet(ABCP), Tet(ABCD) で示す。なお、前者は多面体内の、後者はその外部の四面体とし、Tri(ABC) がそれらに共有する三角形である。もし、Sph(ABCP) がその内部に点 D を包含したとしよう。そうすると図-2 に示したように点 P もまた Sph(ABCD) に包含されることになり、Tet(ABCD) がデラウニー分割であるという仮定に反することになる。したがって、Sph(ABCP) の多面体外領域は必ず Sph(ABCD) の内部に包含され、決して点 D を内部に含むことはない。

上に示した数学的な結果は「いまデラウニー分割が与えられたとき、そこに一点を追加したとき得られる新しいデラウニー分割はその点を包含する外接球をもつ四面体だけを取り出し、それらが構成する多面体の表面上に位置する節点と新たに設定した点 P を結合することにより得られる」ことを示している。

3. アルゴリズム設計

(1) 概 説

前節で得られた結果より、 n 個の節点を用いたデラウニー分割が与えられれば、 $(n+1)$ 個の節点での新しいデラウニー分割は次のようにして求められる。

- ステップ1 新しい点 P を包含する外接球をすべて探す。
- ステップ2 それら外接球をもつ四面体の共有三角形を取り除き多面体を作る。
- ステップ3 多面体上の三角形と P を結び、四面体を作成する。

ここに示した手法を具体化する段階でさまざまな道具の開発が要求される。以下においてそれらをまとめる。

(2) スケーリング

要素分割の対象となる系の大きさはさまざまであり、それらを 1 つの要素分割法で処理するために、ここでは

x, y, z それぞれの方向への対象系の最大寸法をまず求め、その最大長さを単位長さ 1 となるようにする。いま、三軸方向への節点分布長をそれぞれ $L(x), L(y), L(z)$ とすると

$$L_{\max} = \max[L(x), L(y), L(z)] \dots \dots \dots (2)$$

修正された長さは

$$\left. \begin{aligned} S(x) &= L(x)/L_{\max} \\ S(y) &= L(y)/L_{\max} \\ S(z) &= L(z)/L_{\max} \end{aligned} \right\} \dots \dots \dots (3)$$

となる。この操作の後、直交座標 $O-xyz$ の原点近傍にすべての節点が配置されるように座標値を決定する。

(3) スーパー・テトラヘドロンの設定

スケーリングされた節点をすべてその内部に包含し得る四面体（これをスーパー・テトラヘドロンと呼ぶ）を上記座標系内に設定する。上に提案したデラウニー分割では四面体分割された空間内に一点ずつ新たに節点を追加し、デラウニー分割を修正・更新する手法を示している。すなわち、四面体集合の内部に新しく一点を追加すればそれは必ずいずれかの四面体内部に位置することになり、その四面体を手がかりとして修正されなければならない四面体を探し、修正を行うことになる。スーパー・テトラヘドロンの導入により、第 1 番目の節点の設定から上と同様にとり行うことができ、アルゴリズムの統一が可能となる。なお、ここではスーパー・テトラヘドロンの外接円半径を 10 とし、その中心を原点 O に置いている。

(4) 節点順序の入れ替え

上に示したデラウニー三角分割では節点を 1 個ずつ設定し、その点を含む四面体をまず探し出さなければならない。いま第 $n+1$ 点を設定したときを考えよう。もしその点とその直前に設定された第 n 点の近傍に設定され、そして四面体探査が第 n 点を設定したとき新たに生成された四面体の一つより始められるとすれば、第 $n+1$ 点を含む四面体と探査を開始した四面体の間にはそれほど多く四面体が存在するとは考えられない。すなわち、四面体探査に要する演算時間の節約が期待できる。このことはデラウニー三角分割を行うための節点の設定順序が演算時間に影響を及ぼすことを示している。したがって、この最適な節点設定順序を決めるためにビン・ソーティング (bin sorting) 法と呼ばれる方法を導入する。

文献 5) において二次元でのビン・ソーティング法が紹介されている。この方法は節点順序の入れ替え法であり、設定された節点群の支配する空間をビンと呼ばれる小さな正方形の集合に分割し、おのおののビンに与えた順序に従って節点を順序づけ、その順序に従って節点を空間に設定し、デラウニー三角分割を行っている。本研

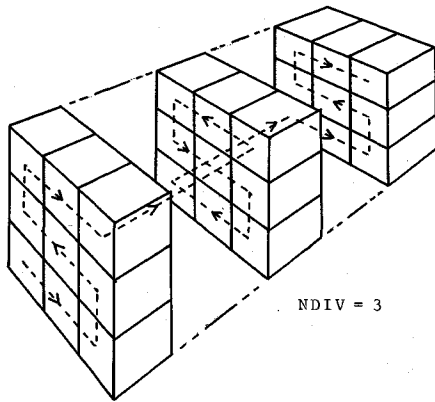


図-3 ビン・ソーティング法

究ではこの二次元でのビン・ソーティング法を次のように三次元に拡張する。

三次元空間に設定された節点群の支配する空間を一辺上の分割数が下式で示される立方体群に分割する。

$$NDIV = (N)^{1/d} \dots\dots\dots (4)$$

上式において N は設定された節点数を示す。以下ではこの小立方体をビンと呼ぶことにする。このようにして作成されたビンに対して図-3中の矢印に示す順序を与え、このビン番号の順序に従っておのおののビンに含まれる点を順次空間に設定し、四面体分割を行うことにする。なお、 d の値はデラウニー三角分割に要する演算時間に影響を与えることが予想されることより、数値実験により最適な d の値を決定することにする。

(5) 追加点によって修正される四面体の探査法

新しい節点の追加によって修正される四面体はまずその節点を含む四面体を探し、次いでその四面体に隣接する四面体を順次探す。この三次元空間の四面体分割アルゴリズムで最も計算時間を要求する部分はこの修正されねばならない四面体の探査法である。よって、ここでは次のような手法を用いて演算時間の短縮を図る。

a) 新しい節点を含む四面体の探査法

すべての四面体のうち、節点はその内部に包含する四面体だけが次の性質を有していることより該当する四面体を探ることができる。いま、任意に1個の四面体を取り出し、その頂点である4個の節点と新しく設定した節点とで4個の小四面体を作成する。それら4個の小四面体の体積の和が元の四面体の体積に一致すれば、その四面体が新しく設定した点を含む。

この探査においてはすでに前段階までに形成されている四面体について逆順に探す。その理由は新しい節点は一般にお互いに近傍に位置することが多いことによる。

b) 修正される四面体の探査法

新たな節点の設定により修正されなければならない四面体を探手段として、四面体要素間の隣接関係を用い

る。まず、a) で探した四面体に隣接する四面体を対象として、それらの外接球の半径の二乗が(外接球中心座標と新しい節点座標の差の二乗)より小さな四面体を見つける。該当する四面体が見つければさらにその四面体をもとにして隣接関係を用いて同様に探す。もし該当する四面体が無ければそれ以上の探査は行わない。このようにして見つけ出された四面体だけが修正される。

(6) デジェネラシー (degeneracy) の扱い

デジェネラシーとは新しい節点を配置した段階で最適な四面体が同時に複数個発生する場合をいう。理論的にはそれらのうちのいずれを採用しても良いことは明らかである。ここではもしこの現象が発生した場合、順序的に最後に見出された四面体を新しい四面体として認識するように設定している。

4. 三次元凸体の自動要素分割法

(1) 入力データと出力データ

入力データは節点数とそれぞれの x, y, z 座標値である。また、出力データは作成された四面体要素数、四面体要素-節点関係、四面体間の隣接関係、外接球の中心座標値である。なお、実際の有限要素解析に必要なデータは入力データと出力データのうちの前者だけである。

その他有限要素解析に必要な出力データ、たとえばある面上に位置する節点番号等については上に示したデータから容易に作成できる。

(2) 要素分割法

ステップ1 データの入力

節点数、節点の座標値を作成し、それらを入力する。

ステップ2 スーパー・テトラヘドロンの設定

上で述べたようにスーパー・テトラヘドロンを構成する4個の節点を空間内に設定する。その後、入力データを用いて、節点群の x, y, z それぞれの軸方向への占有領域のうちの最大値を計算し、節点群が占有する空間の最大長を1となるようにすべての節点の座標を変更し、さらにそれがスーパー・テトラヘドロンの中心部に位置するように設定する。

ステップ3 ビン・ソーティングによる点順序の並べ替え

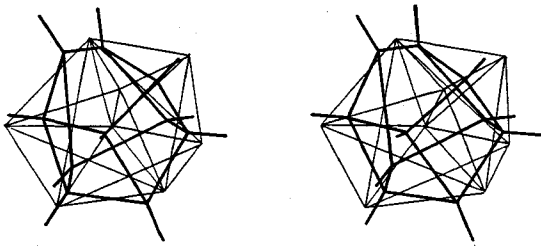
ステップ4 修正される四面体の探査

一つの節点を取り出し、それを包含する四面体を探す。さらに、その節点を包含する外接球をすべて拾い出す(その手法については前節で述べたとおりである)。

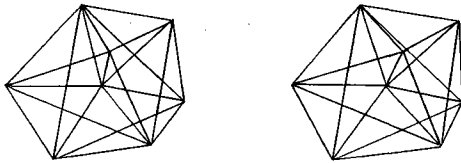
ステップ5 新たな四面体分割

ステップ4で拾い上げられた四面体集合において互いの接線面となっている三角形を取り除き多面体を作り、その表面を覆う三角形と新しく設置した節点とで四面体を再度作成する。

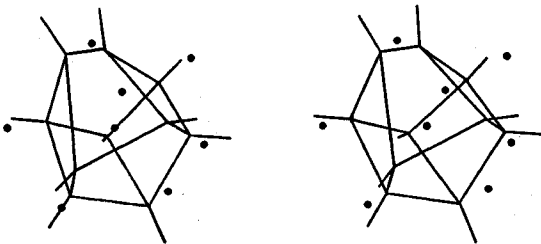
「ステップ4と5を節点数だけ繰り返す。」



(a) ボロノイ分割とデラウニー分割の実体視



(b) デラウニー分割の実体視



(c) ボロノイ分割の実体視

図-4

ステップ6 作成されたデータの修正

作成された四面体要素-節点関係からスーパー・テトラヘドロン上の4個の節点を含む四面体のデータを取り除く。

(3) 結果の確認法

三次元空間の四面体分割がなされた後、実際に空間が四面体でうまく分割できているか不安が残る。三次元空間の分割ではその分割結果を視認することはほとんど不可能に近い。特に本研究で示したように節点配置に関してなんらの法則性をもたない場合、視覚での結果のチェックは不可能である。ここに示した手法は数学的に証明された結果をもとに作成されていることにより、理論的には結果のチェックは不要である。しかしながら、アルゴリズムの中で実数計算（距離計算、体積計算）を行っていることより、数値誤差に起因するエラーが発生することも考えられる。したがって、本手法の適用後次のような方法により結果の確認を行っている。

- a) 四面体が最適であるかどうかの確認法
作成された四面体についてその外接球内部に他の節点を含んでいないかどうか確認する。
- b) 三次元空間が四面体で満たされているかどうかの

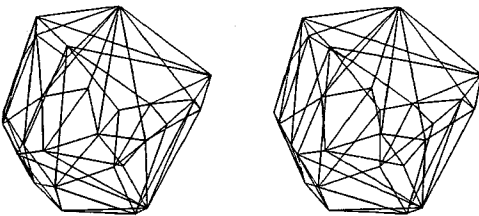
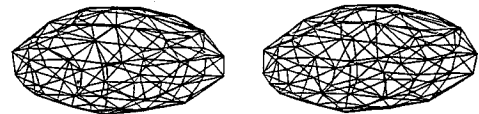
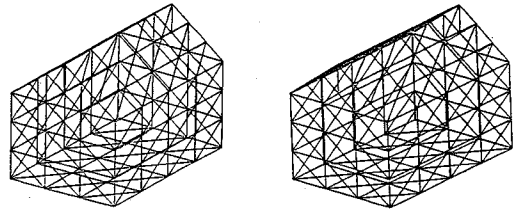
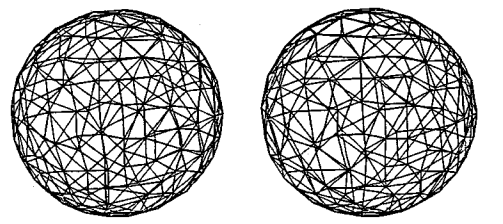


図-5 三次元空間のデラウニー分割例
(表面の三角形分割)

確認

2個の四面体が面接合していれば同じ番号を有する面が必ず2個発生することになる。一方外部に接する面はただ1個しか発生しない。この性質を利用して面の和の計算を行い、もし外部を覆う面だけが残ればその分割は正しいことが判定できる。

(4) 例題

4.(2)で示した要素自動分割法を用いた事例を示す。まず最初の例題でボロノイ分割とデラウニー分割が相対関係にあることを具体的に示そう。図-4に示したのは節点数8個の簡単な分割結果であり、実体視像を用いている。この左・右の像を実体視すればボロノイ分割とデラウニー分割を立体的に視認できよう。(a)はこれら両者を同時に示したものである。(b),(c)はそれぞれデラウニー分割、ボロノイ分割の結果である。この例

表一 ビン・ソーティング法の検討 (1)

節点数 d	1000 (8311)	5000 (32658)	10000 (86001)	15000 (99401)	20000 (132807)	25000 (186548)	30000 (200020)	35000 (238673)	40000 (287231)
4	1.281(6)	—	—	—	—	—	—	—	—
5	1.248(4)	6.853(5)	14.52(6)	22.31(7)	29.78(7)	—	—	—	—
6	1.203(3)	6.795(4)	14.31(5)	21.74(5)	29.00(5)	36.43(5)	44.53(6)	51.87(6)	59.68(6)
7	1.203(3)	6.733(3)	13.90(4)	21.28(4)	28.78(4)	36.06(4)	43.88(4)	52.08(4)	58.91(5)
8	1.177(2)	6.733(3)	13.90(3)	21.13(3)	28.54(3)	36.06(4)	43.88(4)	52.08(4)	59.21(4)
9	1.177(2)	6.733(3)	13.90(3)	21.13(3)	28.54(3)	35.92(3)	43.37(3)	50.69(3)	58.40(3)
10	1.177(2)	6.667(2)	13.90(3)	21.13(3)	28.54(3)	35.92(3)	43.37(3)	50.69(3)	58.40(3)
11	1.177(2)	6.667(2)	13.90(2)	21.21(2)	28.74(2)	35.92(3)	43.37(3)	50.69(3)	58.40(3)
12	1.177(2)	6.667(2)	13.90(2)	21.21(2)	28.74(2)	36.24(2)	44.38(2)	52.70(2)	60.47(2)
×	1.203	6.930	14.58	22.61	30.66	38.99	47.51	58.12	64.81

秒 (辺上のピン数)

表二 ビン・ソーティング法の検討 (2)

bin	探索時間 LOCATE(s)	分割時間 POLY(s)	演算時間 TOTAL(s)
11	0.6295	18.93	23.57
7	0.7862	17.50	22.31
5	0.9499	16.71	21.74
4	1.122	16.05	21.28
3	1.429	15.54	21.13
2	2.056	14.93	21.21
×	3.933	14.43	22.53

(節点数 15000 要素数 99401)

題から三次元空間が四面体とその相対である 8 個の多面体分割ができていることが理解できよう。

図-5 はさまざまな三次元空間の分割例である。それらは多くの節点を有していることから、図としてはそれらの表面の三角形分割だけを取り出し、それを実体視像で示している。いずれの場合も適切に分割されていることが確認できよう。なお、ここに示した面の表現は 4.(3) で示した確認法を図化したものである。

次に本研究で提案したデラウニー三角分割に要する演算時間の検討を行う。文献 4) では、デラウニー三角分割に要する演算時間は、生成された要素数を M としたとき、 $O(M \log M)$ となる。すでに述べたようにここで提案したデラウニー三角分割ではビン・ソーティング法を導入して演算時間の短縮を図っている。このビン・ソーティング法では式 (4) に示される d を決定しなければならない。この目的のため立方体を対象としてその表面および内部に乱数を用いて節点を発生させ、それら節点をビン・ソーティング法により節点順序を付け替

えた後、デラウニー法を適用する。表一は d の値を変化させた場合の演算時間の比較を行ったものである。対象とした問題は 1000 点から最大 40000 点までであり、デラウニー三角分割にはスーパーコンピュータ (SX-1E) を使用した。結果として、 $d=10$ の場合、デラウニー三角分割に要する演算時間が最小となっていることがわかる。なお、同図中、節点数の下の () は生成された要素数を、そして、演算時間の後の () は一辺のビン分割数を示している。また、最下行はビン・ソーティング法を利用しない場合の演算時間を示す。

表二は一辺上のピン数を変化させたとき、点を含む四面体の探索時間の合計 (LOCATE) と新しく設定された点を外接球内に含む四面体集合で作られる多面体よりデラウニー三角分割を行うのに要した演算時間の和 (POLY), そして、それらの合計の演算時間 (TOTAL) を示したものである。この表より、ピン数が増加するに従い LOCATE が減少し、一方、POLY が増加することがわかる。すなわち、ある個数のピン数 (この数値実験では $d=10$ と設定して得られるピン数) に分割したとき、その和としてのデラウニー三角分割の演算時間が最小となることが示される。

これらの演算時間と節点数との関係を図-6 に示す。この図より、本論文で提案したデラウニー三角分割法は、少なくとも 4 万点までにおいては、その演算時間は節点数にほぼ比例していると考えられ、ここに提案したデラウニー三角分割アルゴリズムの有効性が示されたといえよう。同様の関係は図-7 においてもいえ、演算時間はほぼ生成された要素数に比例して増加している。

ビン・ソーティングを採用したとき、採用しない場合に比べて約 10% 程度の演算時間が節約され、この差はビン・ソーティングの効果である。一方、文献 5) では

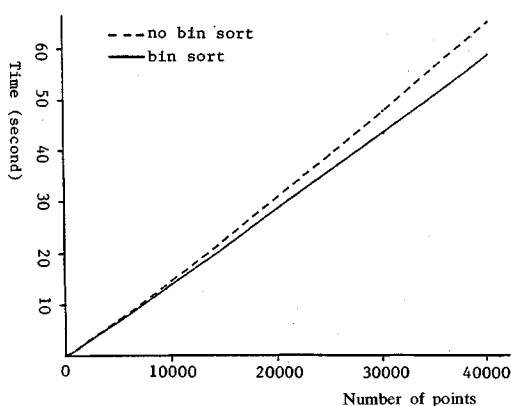


図-6 節点数と演算時間の関係

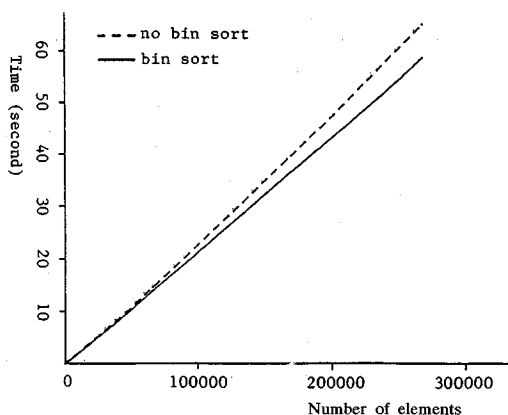


図-7 要素数と演算時間の関係

二次元においてピン・ソーティング法は非常に有効であることが示されている。この差は、二次元では、1点を追加して新しいデラウニー三角分割を求める段階で、2個の三角形より作られる四辺形の対角線の比較を行い、短い方の対角線を用いて2個の三角形に分割する、いわゆるスワッピング・アルゴリズムを利用できるが⁶⁾、本研究で扱っている三次元の場合、このようなスワッピングが利用できないことによる。三次元においてスワッピングが利用できない理由は、たとえ1つの四面体の外接球の中に他の四面体の頂点が含まれた場合でも、これら2個の四面体を組み合わせてできる六面体が凹六面体となる場合が生じ、四面体分割が不可能となることによる。しかしながら、図-6, 7をみてもわかるように、ここに提案したデラウニー三角分割の演算時間は、節点数、あるいは生成された要素数とほぼ線形な関係にあり、非常に高速であるといえよう。

5. あとがき

本研究においてはボロノイ分割を用いた三次元凸空間の四面体分割法を提案し、数値実験によりその有効性を検証した。この数値実験よりここに示された方法は、節点数あるいは生成された要素数にほぼ比例した演算時間しか要求しない高速なものとなっていることが示された。この方法によって作成された四面体は幾何学的に最適な形状を保証するものであり、そのまま三次元有限要素解析のモデル設定に利用できる。また、節点配置に関しては何らの拘束もないことより、一般的な三次元体の有限要素解析モデルの設定はいうまでもなく、三次元亀裂伝播解析のように幾何学的に複雑な形状を対象としてモデル化しなければならない場合等においても非常に有効な要素分割法である。

本研究のもととなっているボロノイ分割とその相対問

題であるデラウニー分割ではともに節点の設定だけが入力データとして要求される。言い換えると三次元空間の部分空間を定義する境界という概念はそこには存在しない。したがって、境界領域の幾何学的概念である凸・凹は認識できず、四面体分割が不必要な凹部も含めた三次元部分空間を四面体群で満たしてしまう結果になる。今後の課題としてこの“へこんだ部分”をいかにして認識させるか、そしてその部分を除外した空間内だけに限定してどのようにボロノイ分割・デラウニー分割を行うかが挙げられる。この問題を解決して初めて任意形状を有する三次元体の要素自動分割法となり得るといえよう。

参考文献

- 1) Yerry, M.A. and Shephard, M.S. : Automatic three-dimensional mesh generation by the modified octree technique, *Int. J. Num. Meth. Eng.*, Vol.20, pp.1965~1990, 1984.
- 2) Bowyer, A. : Computing Dirichlet tessellations, *The Computer Journal*, Vol.24, No.2, pp.162~166, 1981.
- 3) Watson, D.F. : Computing n -dimensional Delaunay tessellation with application to Voronoi polytopes, *The Computer Journal*, Vol.24, No.2, pp.167~172, 1981.
- 4) Baker, T.J. : Generation of tetrahedral meshes around complete aircraft, *Numerical Grid Generation in Computational Fluid Mechanics '88* (ed. Sengupta, S., Hauser, J., Eiseman, P.R. and Thompson, J.F.), pp.675~685, 1988.
- 5) Sloan, S.W. : A fast algorithm for constructing Delaunay triangulations in the plane, *Advances in Engineering Software*, Vol.9, No.1, pp.34~55, 1987.
- 6) Lawson, C.L. : Software for C^1 interpolation, in Rice, J. (ed.) *Mathematical Software*, Academic Press, New York, pp.161~194, 1977.

(1989.12.27 受付)

AUTOMATIC MESH GENERATION OF 3-DIMENSIONAL CONVEX DOMAIN

Takeo TANIGUCHI and Chikashi OHTA

The aim of this investigation is the proposal of new automatic mesh generation method for 3-dimensional convex domain. The strategy is based on the Voronoi tessellation, and the dual relation between the Voronoi tessellation and the Delaunay triangulation is used to decompose arbitrary 3-D convex domain into tetrahedra with optimum geometrical configuration. Proposed method is directly applied to the mesh generation method for 3-D finite element analysis. This paper discusses the mathematical background of the mesh generation method, the design of the automatic mesh generation, the details of the mesh generation procedure, and the results of numerical examples. At present the method can be applied only for 3-D convex domain.
