

# **ADVENTURE\_TetMesh**

**Automatic generation of tetrahedral mesh from triangular surface patches**

**Version : b - 0.91**

**プログラム使用マニュアル**

**October 05 , 2005**

**ADVENTURE Project**

## 目 次

1. 概要.....	1
2. 稼動環境.....	2
3. インストールについて.....	2
3.1 インストール方法.....	2
3.2 ディレクトリ構成の説明.....	2
4. 四面体メッシュ作成プログラム操作説明.....	3
4.1 プログラムの操作フロー.....	3
4.2 プログラム実行例（単一材料）.....	6
4.2.1 プログラムの実行.....	6
4.2.2 実行ログ.....	7
4.2.3 実行結果.....	7
4.3 プログラム実行例（複数材料）.....	10
4.3.1 プログラムの実行.....	10
4.3.2 実行ログ.....	10
4.3.3 実行結果.....	11
4.4 コマンドオプション.....	15
4.4.1 TetMesh_P のコマンドオプション.....	15
4.4.2 TetMesh_M のコマンドオプション.....	17
4.4.3 TetMesh_S のコマンドオプション.....	17
5. 四面体メッシュ評価プログラム.....	18
5.1 四面体メッシュ評価プログラムの実行.....	18
5.1.1 プログラムの実行例.....	18
5.1.2 プログラム実行結果.....	18
5.1.3 プログラムの実行ログ.....	19
5.2 プログラムのコマンドオプション.....	20
6. ファイル仕様.....	21
6.1 表面パッチデータファイル.....	22

6.2 節点密度制御ファイル .....	23
6.3 メッシュデータファイル.....	26
参考文献.....	28
付録 .....	29

## 1. 概要

本プログラムは、三角形表面パッチを入力し Delaunay 四面体分割により四面体メッシュを作成します。本プログラムは、入力した表面パッチから平滑化と Delaunay 化して表面メッシュを作成するモジュール **TetMesh\_P**、Delaunay 四面体分割により四面体メッシュを作成するモジュール **TetMesh\_M**、四面体メッシュに二次節点を追加して二次要素にするモジュール **TetMesh\_S** の3つのモジュールで構成されています。

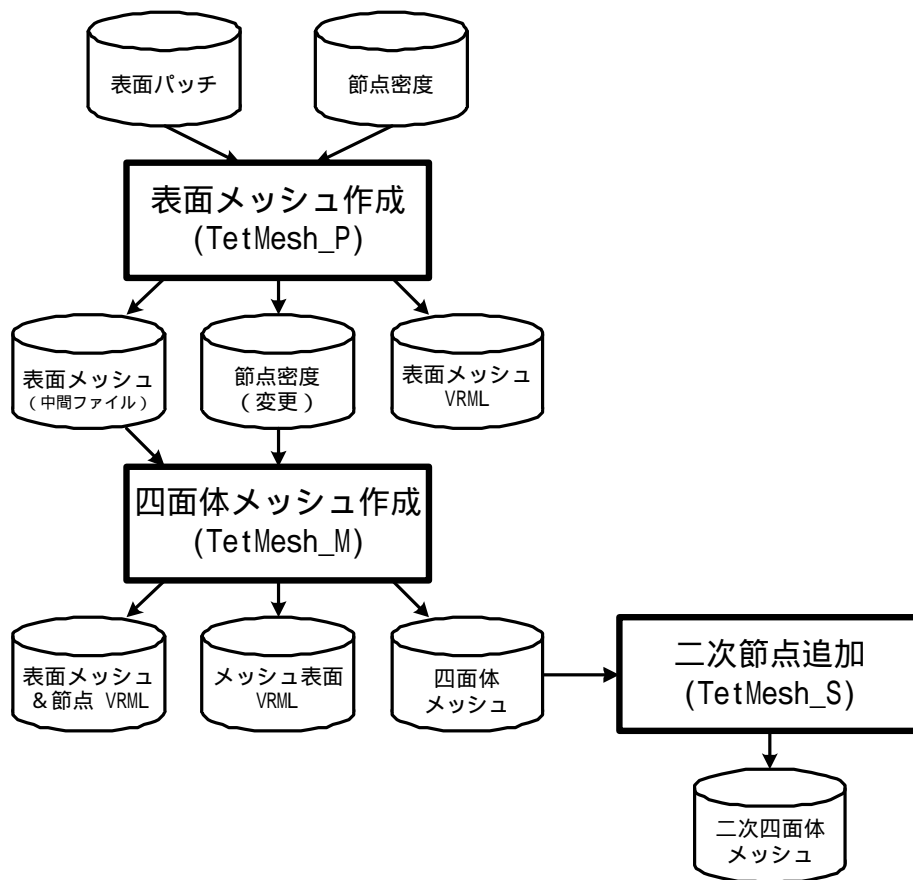
また、作成された四面体メッシュの評価を行うツール **TetMesh\_E** が付属しています。作成された四面体メッシュの情報は、以下のファイルに出力されます。

- (1) 四面体メッシュデータファイル (拡張子: **msh**)

四面体メッシュの節点座標, 要素コネクティビティおよび材料毎の要素番号

- (2) 表面 VRML ファイル (拡張子: **wr1**)

メッシュ表面を VRML 形式に変換したデータ (2種類)



ADVENTURE\_TetMesh

## 2. 稼動環境

本プログラムは以下の環境で動作確認しています。

### (1) OS

UNIX, Linux

### (2) Compiler

TetMesh\_P : Fortran90 (DIGITAL Fortran 90 V5.2-705,

PGI Fortran 90 V3.2-3,

g95 (Sep 25 2005) で動作確認)

TetMesh\_M : C++ (Compaq C++ V6.2-024,

g++ (Ver 2.9x, 3.x, 4.0.1) で動作確認)

TetMesh\_S : C++ (Compaq C++ V6.2-024,

g++ (Ver 2.9x, 3.x, 4.0.1) で動作確認)

TetMesh\_E : C++ (Compaq C++ V6.2-024,

g++ (Ver 2.9x, 3.x, 4.0.1) で動作確認)

## 3. インストールについて

### 3.1 インストール方法

tar + gz 形式のモジュールを展開し、トップディレクトリにある `INSTALL.jp` の内容に従ってインストールしてください。

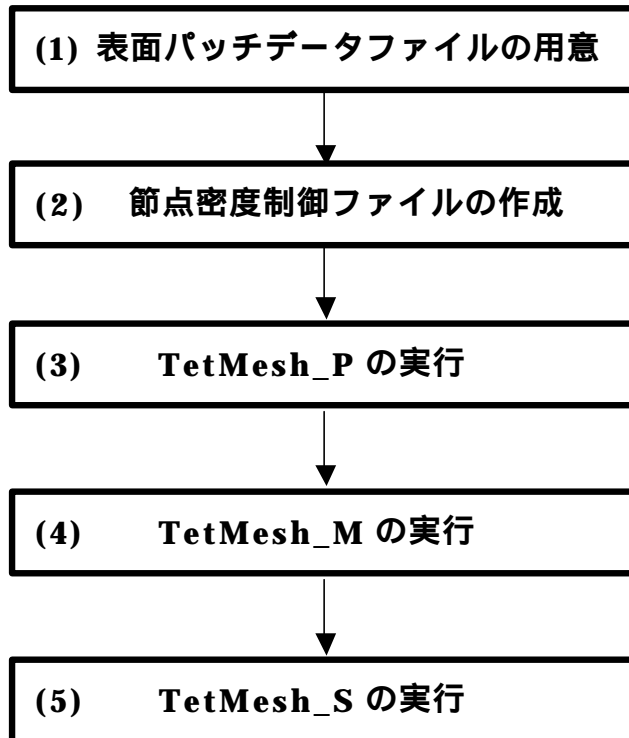
### 3.2 ディレクトリ構成の説明

トップディレクトリ直下にある `README.jp` を参照してください。

## 4. 四面体メッシュ作成プログラム操作説明

### 4.1 プログラムの操作フロー

プログラムの実行フローを以下に示します。



#### (1) 表面パッチデータファイルの用意

- ・表面パッチデータは「6.1 表面パッチデータファイル」に示すフォーマットに従って作成されている必要があります。
- ・使用するファイルは、ADVENTURE\_TriPatch の出力を想定しています。
- ・ファイルの拡張子は pcm とします。
- ・単一材料であれば旧フォーマット (pch) でも入力可能です。

#### (2) 節点密度制御ファイルの作成

- ・節点密度制御ファイルは「6.2 節点密度制御ファイル」を参照して作成して下さい。
- ・ADVENTURE\_TriPatch の出力を使用する場合は、表面パッチ作成時に使用した節点密度制御ファイルをそのまま使用してください。

- ・入力する表面パッチデータファイルと拡張子を除いて同じ名前である必要があります。
- ・ファイルの拡張子は `ptn` とします。

### (3) TetMesh\_P の実行

TetMesh\_P は表面パッチを入力し、内部メッシュ生成のための表面メッシュを生成します。以下のコマンドで実行します。

```
advtmesh9p 表面パッチデータファイル名 -d
```

ただし、表面パッチデータファイル名は拡張子を除いて入力してください。

コマンドオプションの `-d` は、節点密度制御ファイルを使用することを指定するためのものです。コマンドオプションについては後で説明します。

TetMesh\_P の実行により、表面メッシュデータ中間ファイル（拡張子は `pcc`）と節点密度制御ファイル（拡張子は `ptn`）が出力されます。それぞれもとのファイル名の後に `"c"` を付加したファイル名となります。

必要であれば、コマンドオプションの `-p` を指定することにより作成した表面メッシュを VRML 形式（VRML フォーマット Ver 1.0）に変換したファイルを出力し、お手持ちの VRML ブラウザで表示させることができます。ファイル名は指定した表面パッチデータファイル名に `_c.wrl` を付加したものになります。

### (4) TetMesh\_M の実行

TetMesh\_M は TetMesh\_P が出力した表面メッシュと節点密度制御ファイルを入力し、内部に四面体メッシュを生成します。以下のコマンドで実行します。

```
advtmesh9m 表面メッシュデータ中間ファイル名
```

ただし、表面メッシュデータ中間ファイル名は拡張子を除いて入力してください。

コマンドオプションについては後で説明します。

TetMesh\_M の実行により、四面体メッシュデータファイルが出力されます（出力される四面体メッシュは一次要素です）。

四面体メッシュデータファイルの拡張子は `msh` です。

必要であれば、コマンドオプションの `-p` を指定することにより作成した四面体メッシュの表面を VRML 形式（VRML フォーマット Ver 1.0）に変換したファイルを出力し、

お手持ちの VRML ブラウザで表示させることができます。ファイル名は指定した表面メッシュデータファイル名に `_n.wrl` を付加したものと `_e.wrl` を付加したものの2種類です。

本プログラムの実行により出力された四面体メッシュデータファイルは、ADVENTURE\_BCtool の入力として利用されます。

#### (5) TetMesh\_S の実行

TetMesh\_S は TetMesh\_M が出力した四面体メッシュ（一次要素）を入力し、二次四面体メッシュを生成します。以下のコマンドで実行します。

**advtmesh9s**    **一次四面体メッシュデータファイル名**

ただし、一次四面体メッシュデータファイル名は拡張子を除いて入力してください。コマンドオプションについては後で説明します。

TetMesh\_S の実行により、二次四面体メッシュデータファイルが出力されます。ファイル名はもとのファイル名の後に “s” を付加したものとなります。

四面体メッシュデータファイルの拡張子は一次、二次とも `msh` です。

本プログラムの実行により出力された四面体メッシュデータファイルは、ADVENTURE\_BCtool の入力として利用されます。



## 4.2 プログラム実行例（単一材料）

### 4.2.1 プログラムの実行

トップディレクトリの下に `sample_data` にサンプルデータが格納されています。ここでは `adventure_manual_data01.pcm` および `adventure_manual_data01.ptn` を用いて実行例を示します。

#### (1) TetMesh\_P の実行

```
% advtmesh9p  adventure_manual_data01  -d  -p
```

これにより `adventure_manual_data01.pcm`, `adventure_manual_data01.ptn` を入力し, `adventure_manual_data01c.pcc`, `adventure_manual_data01c.ptn` および `adventure_manual_data01_c.wrl` の3つのファイルが作成されます。

#### (2) TetMesh\_M の実行

```
% advtmesh9m  adventure_manual_data01c  -p
```

これにより `adventure_manual_data01c.pcc`, `adventure_manual_data01c.ptn` を入力し, `adventure_manual_data01c.msh`, `adventure_manual_data01c_n.wrl` および `adventure_manual_data01c_e.wrl` の3つのファイルが作成されます。

#### (3) TetMesh\_S の実行

```
% advtmesh9s  adventure_manual_data01c
```

これにより `adventure_manual_data01c.msh` を入力し, `adventure_manual_data01cs.msh` ファイルが作成されます。

#### 4.2.2 実行ログ

上記の例を実行すると出力されるメッセージログの説明を付録 A , B および C に示します .

#### 4.2.3 実行結果

本プログラムの実行により出力される VRML ファイル (VRML フォーマット Ver 1.0) をお手持ちの VRML ブラウザによって見ることができます .

##### (1) 入力パッチ

入力したもともとの表面パッチは以下のコマンドを実行することにより VRML に変換できます .

```
% advtmesh9p  adventure_manual_data01  -cr  -p
```

これにより `adventure_manual_data01_c.wrl` というファイルが作成されます .  
-cr オプションを指定すると , 入力表面パッチを VRML に変換するだけで表面メッシュの作成は行ないません .

ここで出力された VRML ファイルをブラウザで表示した例を下図に示します .

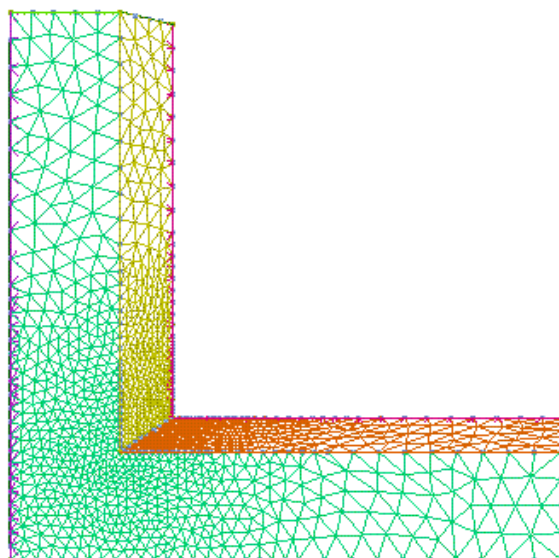


図 4.2.3-1 入力表面パッチの VRML 表示例

## (2) 表面メッシュ

4.2.1 (1) TetMesh\_P の実行により作成した表面メッシュと同時に出力された VRML ファイルをブラウザで表示した例を下図に示します。

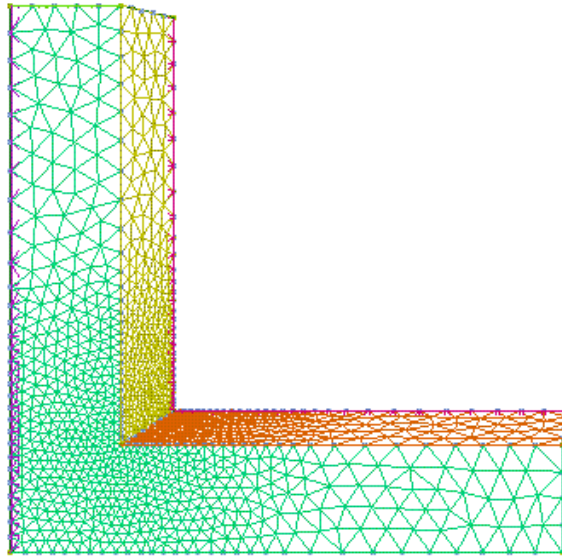


図 4.2.3-2 表面メッシュの VRML 表示例

## (3) 四面体メッシュの表面

4.2.1 (2) TetMesh\_M の実行により作成した四面体メッシュの表面はオリジナル表面パッチのファイル名の後ろに `c_e.wr1` が付いた VRML ファイルを表示することにより見ることができます。

また、四面体メッシュの節点はオリジナル表面パッチのファイル名の後ろに `c_n.wr1` が付いた VRML ファイルを表示することにより見ることができます。表示している点は、赤が表面節点（表面メッシュの頂点に一致）、水色が内部節点を表わします。

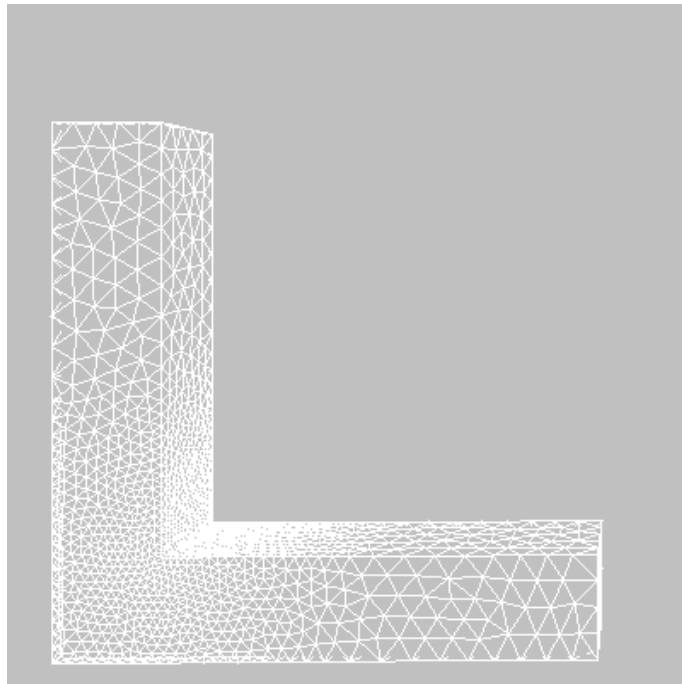


図 4.2.3-3 四面体メッシュの表面の VRML 表示例

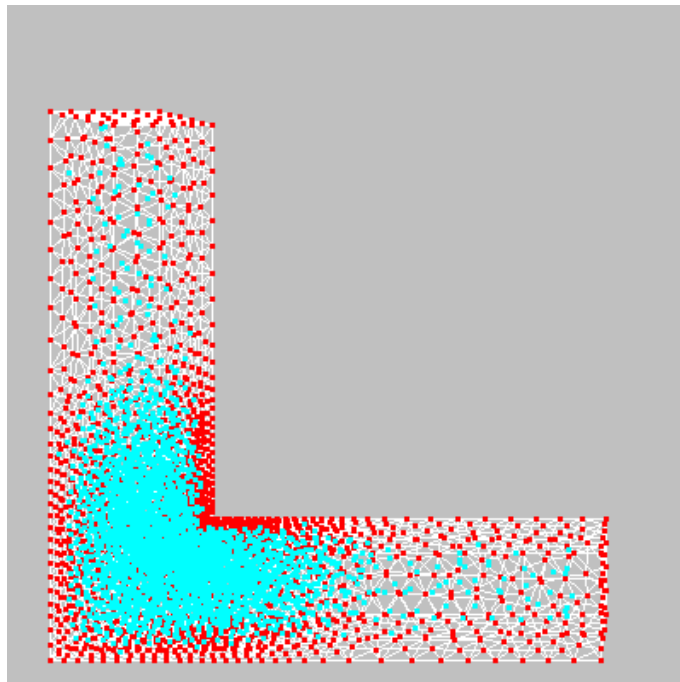


図 4.2.3-4 四面体メッシュの節点の VRML 表示例 (ワイヤフレーム表示)

## 4.3 プログラム実行例（複数材料）

### 4.3.1 プログラムの実行

トップディレクトリの下に `sample_data` にサンプルデータが格納されています。ここでは `mat_in0102.pcm` および `mat_in0102.ptn` を用いて複数材料の場合の実行例を示します。

#### (1) TetMesh\_P の実行

```
% advtmesh9p mat_in0102 -d -p
```

これにより `mat_in0102.pcm`, `mat_in0102.ptn` を入力し, `mat_in0102c.pcc`, `mat_in0102c.ptn` および `mat_in0102_c.wrl` の3つのファイルが作成されます。

#### (2) TetMesh\_M の実行

```
% advtmesh9m mat_in0102c -p
```

これにより `mat_in0102c.pcc` `mat_in0102c.ptn` を入力し `mat_in0102c.msh`, `mat_in0102c_n.wrl` および `mat_in0102c_e.wrl` の3つのファイルが作成されます。

#### (3) TetMesh\_S の実行

```
% advtmesh9s mat_in0102c
```

これにより `mat_in0102c.msh` を入力し, `mat_in0102cs.msh` ファイルが作成されます。

### 4.3.2 実行ログ

上記の例を実行し, 出力されるメッセージログの説明を付録 D, E および F に示しま

す。

### 4.3.3 実行結果

本プログラムの実行により出力される VRML ファイル (VRML フォーマット Ver 1.0) をお手持ちの VRML ブラウザによって見ることができます。

#### (1) 入力パッチ

入力したもともとの表面パッチは以下のコマンドを実行することにより VRML に変換できます。

```
% advtmesh9p mat_in0102c -cr -p
```

これにより `mat_in0102c_c.wr1` というファイルが作成されます。-cr オプションを指定すると、入力表面パッチを VRML に変換するだけで表面メッシュの作成は行いません。

ここで出力された VRML ファイルをブラウザで表示した例を下図に示します。

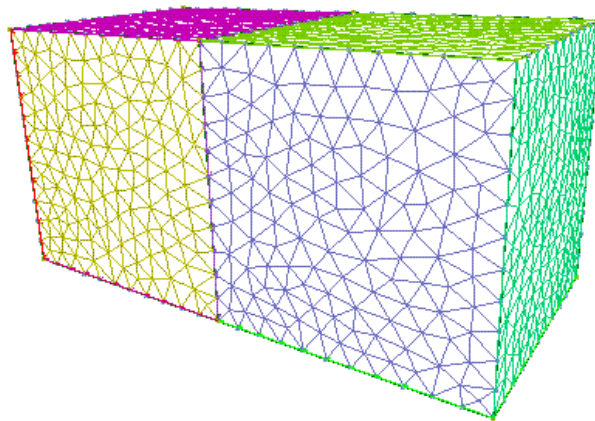


図 4.3.3-1 入力表面パッチの VRML 表示例

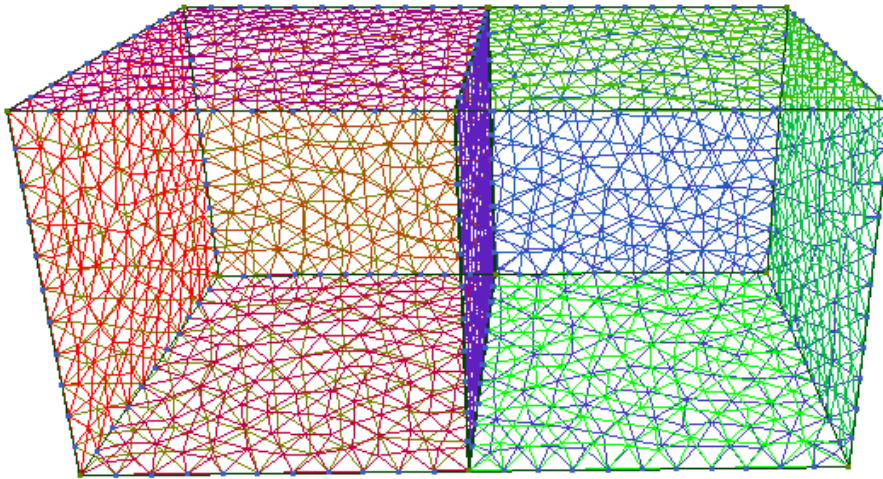


図 4.3.3-2 入力表面パッチの VRML 表示例 (ワイヤフレーム表示)

## (2) 表面メッシュ

4.3.1 (1) TetMesh\_P の実行により作成した表面メッシュと同時に出力された VRML ファイルをブラウザで表示した例を下図に示します。

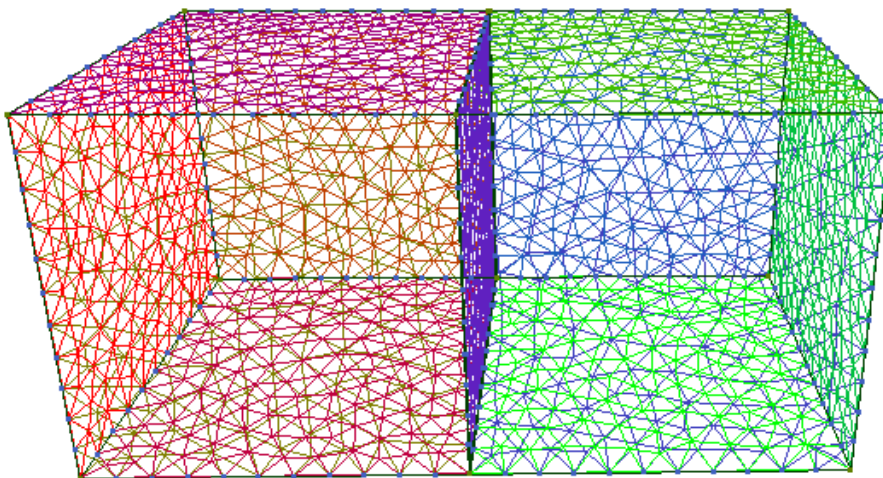


図 4.3.3-3 表面メッシュの VRML 表示例 (ワイヤフレーム表示)

### (3) 四面体メッシュの表面

4.3.1 (2) TetMesh\_M の実行により作成した四面体メッシュの表面はオリジナル表面パッチのファイル名の後ろに `c_e.wrl` が付いた VRML ファイルを表示することにより見ることができます (図 4.3.3-4) .

また, 四面体メッシュの節点はオリジナル表面パッチのファイル名の後ろに `c_n.wrl` が付いた VRML ファイルを表示することにより見ることができます (図 4.3.3-5) . 表示している点は, 赤が表面節点 (表面メッシュの頂点に一致), 水色が内部節点を表わします .

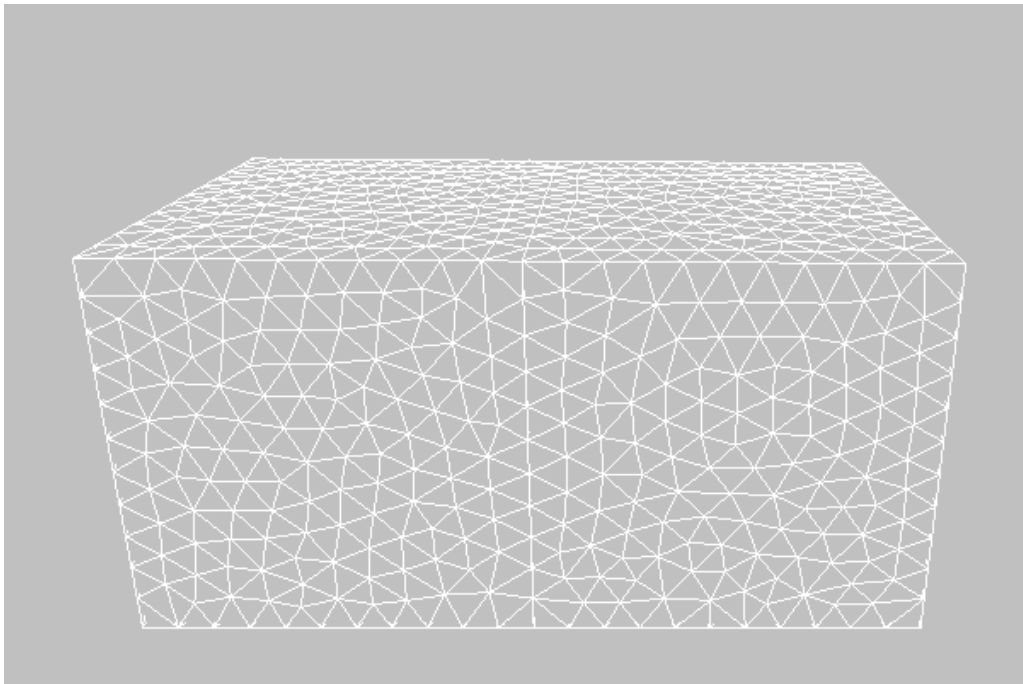


図 4.3.3-4 四面体メッシュの表面の VRML 表示例



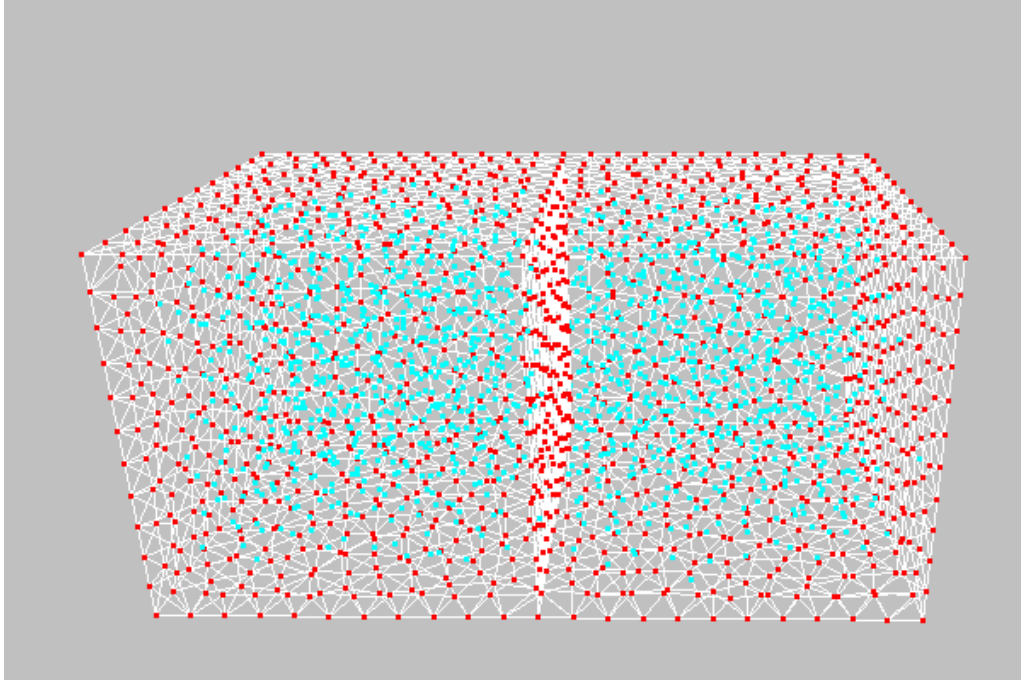


図 4.3.3-5 四面体メッシュの節点の VRML 表示例 (ワイヤフレーム表示)

## 4.4 コマンドオプション

### 4.4.1 TetMesh\_P のコマンドオプション

表面メッシュ作成プログラム TetMesh\_P は、入力した表面パッチに対して平滑化と Delaunay 再分割を同時に行なう Pliant Delaunay Retriangulation の手法を採用しています。平滑化は Bossen and Heckbert<sup>(1)</sup>による分子内 Lennard-Jones ポテンシャルの近似関数を適用する方法と、周囲の点の重心に移動する Laplacian smoothing の方法を併用しています。本プログラムは、入力した表面パッチの頂点を出発点として、上記の平滑化と粗密に従った頂点の追加、削除を行ない、Delaunay 四面体分割を行なった後に表面がそのまま現れるような頂点配置になる表面メッシュを作成します。Delaunay 分割では、同一円周上に 4 点以上の点がある場合 (degeneracy と呼ばれます) には分割に任意性があります。そのため、ここで生成する三角形分割を「表面メッシュ」と呼び、四面体メッシュの表面と区別しています。なお、材料境界では degeneracy が起きない頂点配置にして、材料毎に別々に四面体メッシュを作成しても境界面の三角形要素が一致するようにしています。本プログラムは以下の処理手順で実行されます。

- (1) 表面パッチデータの入力
- (2) 節点密度制御データの入力
- (3) 極端につぶれた要素の削除
- (4) 表面のグループ化
- (5) 入力した頂点を動かさないで Delaunay 再分割
- (6) 細かい形状を検索して節点密度を自動調整
- (7) 節点密度分布に従っておおまかに頂点を追加または削除
- (8) 平滑化と Delaunay 再分割を同時に行なう Pliant Delaunay Retriangulation
- (9) 境界エッジと表面の保護

(8)と(9)は二重のループによる収束計算を行なっています。

以下に本プログラムのコマンドオプションを示します。なお、オプションの後に値を続ける場合は、空白を空けずに指定してください。

- d 節点密度制御ファイルの指定。-d に続けて表面パッチファイル名とは別のファイル名を指定することもできます。ただし、ファイル名を指定する場合は拡張子も含めて指定してください。  
本オプションは -base オプションとは排他的です。
- base 基本節点間隔の指定。-base に続けて基本節点間隔を指定します。  
均質なメッシュまたは上記(6)の自動調整のみでメッシュを作成する場合、節点密度制御ファイルを用意しなくてもこのオプションで指定することがで

きます。これは `-d` オプションを指定して節点密度制御ファイルの中で `BaseDistance` を指定するのと同じです。ただし、節点密度を自動調整すると計算時間が増加しますので、複雑な形状の場合はなるべく節点密度制御ファイルで密度制御することをお勧めします。

本オプションは `-d` オプションとは排他的です。

`-d` も `-base` も指定しなかった場合は、入力した表面パッチの平均エッジ長を基本節点間隔として採用します。

- eh 許容要素高最小値の指定。-eh に続けて 0 ~ 0.2 の範囲で指定します。値は局所的な節点間隔に対する割合で指定します。デフォルトは 0.05 です。極端につぶれた要素（パッチ）がある場合、面の方程式が立てられなかったり、形状による節点密度の自動調整により極端に細かいメッシュが作成されたりすることがあります。そのため、本プログラムでは要素高を局所的な節点間隔で割った値が指定値よりも小さい要素を削除します。ただし、要素を削除することによって形状が悪くなると判断した場合は削除しません。本オプションはあまり大きな値を指定すると計算が破綻する場合がありますので注意が必要です。また、-eh だけを指定すると極端につぶれた要素があっても要素の削除は行ないません。
  
- sm 平滑化オプション。-sm の後に 2 または 3 を指定します。デフォルトは 3 です。本プログラムでは、平滑化の手法として Bossen による方法と Laplacian smoothing を併用します（3 の場合）。まず Bossen の方法で平滑化を行ない、収束したら Laplacian smoothing で再度平滑化します。ただし、形状によっては Bossen で収束しても Laplacian で収束しない場合があります。そのような場合は -sm2 と指定することで Laplacian smoothing を行わないようにすることができます。
  
- cr 入力した表面パッチをそのまま VRML 形式で見たいとき、-p オプションと併せて指定します。この場合、表面メッシュの作成は行ないません。
  
- p[n] VRML ファイル出力オプション。-pn と指定すると VRML ファイルの座標データを規格化して出力します。本プログラムは、入力した表面パッチの角度を判定して面をグループ分けします。本オプションで出力した VRML ファイルは面グループを色分けして表示します。計算が収束しなかった場合、本オプションを指定していると、全体を水色で表示し、収束しなかった点を赤で表示します。ファイル名は指定した表面パッチデータファイル名に `_c.wrl` を付加したものになります。

#### 4.4.2 TetMesh\_M のコマンドオプション

四面体メッシュ生成プログラム TetMesh\_M は, TetMesh\_P で作成した表面メッシュの頂点を表面節点とし, 内部に節点を追加して四面体メッシュを作成します。内部節点の発生にはバケット法, 四面体分割は Delaunay 分割を採用しています<sup>(2)</sup>。本プログラムは以下の処理手順で実行されます。

- (1) 節点密度制御データの入力
- (2) 表面メッシュデータの入力
- (3) 表面節点の発生
- (4) バケット法による内部節点の発生
- (5) Delaunay 分割による四面体要素作成
- (6) 形状外部の要素削除
- (7) 内部スライバー要素の修正

複数領域 (複数材料) の場合, (3) ~ (7) の処理を領域毎に繰り返します。

以下に本プログラムのコマンドオプションを示します。

- p** VRML ファイル出力オプション。本オプションを指定すると, 指定した表面メッシュデータ中間ファイル名に `_n.wrl` を付加したものと `_e.wrl` を付加したものの 2 つの VRML ファイルが出力されます。
- ファイル名に `_n.wrl` が付加された VRML ファイルは, 入力した表面メッシュと発生した節点を表示します。赤が表面節点 (表面メッシュの頂点に一致), 水色が内部節点を表わします。ファイル名に `_e.wrl` が付加された VRML ファイルは, 作成した四面体メッシュの表面を表示します。

#### 4.4.3 TetMesh\_S のコマンドオプション

二次節点追加プログラム TetMesh\_S は, TetMesh\_M で作成した一次四面体メッシュを入力し, 各要素のエッジ中点に二次節点を追加します。

以下に本プログラムのコマンドオプションを示します。

- show** 表示オプション。本オプションを指定すると, 二次節点を追加して二次要素にしたファイルを出力せず, 二次要素にした場合の節点数と自由度数の表示のみを行いません。

## 5. 四面体メッシュ評価プログラム

本プログラムは、四面体メッシュの評価を行いません。評価項目は、エッジ長、二面挟角、辺長換算体積、要素高アスペクト比逆数、最小要素高さ分布です。

四面体メッシュは、一次要素でも二次要素でも評価可能です。

### 5.1 四面体メッシュ評価プログラムの実行

メッシュデータファイルを入力ファイルとして指定します。

```
advtmesh9e   メッシュデータファイル名   -p
```

ただし、メッシュデータファイル名は拡張子を除いて指定してください。

#### 5.1.1 プログラムの実行例

トップディレクトリの `sample_data` のデータを例題として用います。

`mati_in0102cs.msh` は、`mati_in0102.pcm` の入力データから、`TetMesh_P` 及び `TetMesh_M` を実行させ、`TetMesh_S` で 2 次節点を作成した四面体メッシュデータファイルです。

```
advtmesh9e   mati_in0102cs   -p   -d
```

`-p` は、不正または評価基準値以下の要素を VRML で表示するオプションです。また、`-d` は、節点密度との比を計算します。詳しくは、「5.2 プログラムのコマンドオプション」で示します。

#### 5.1.2 プログラム実行結果

上記の実行により、出力された VRML の表示例を図 5.1.2-1 に示します。

この例では、基準値を下回る要素がないため、要素の高さが最小となる要素のみが赤で表示されています。

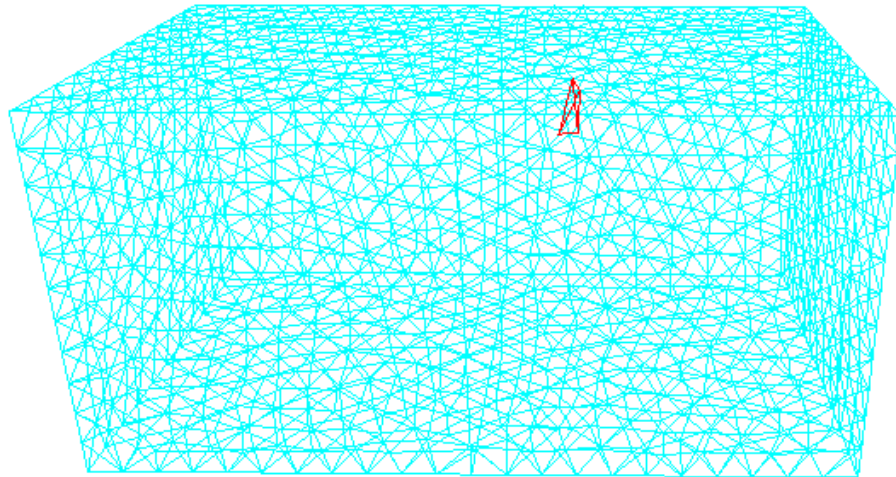


図 5.1.2-1 メッシュの評価 VRML 表示例 (ワイヤフレーム表示)

### 5.1.3 プログラムの実行ログ

上記の例を実行し、出力されるメッセージログの説明を付録 G に示します。

メッシュの評価項目は以下の 5 つです。

<X> は分布の左側の項目 <Y> は右側の項目を表します。密度比分布は、実行時に -d オプションを指定した時に表示されます。

#### (1) エッジ長さ分布

<X> 四面体要素の辺の長さ (実寸)

<Y> 辺の数

密度比分布 ( option )

<X> エッジの 2 端点の節点密度から求めた値の平均で辺の長さを割ったもの

<Y> 辺の数

#### (2) 二面挟角分布

<X> 四面体要素の二面挟角の最大値・最小値

<Y> 要素数

#### (3) 辺長換算体積分布

<X> 辺長換算体積 (要素体積と同体積の正四面体の辺長)

<Y> 要素数

密度比分布 ( *option* )

<X> 要素の重心における節点間隔で辺長換算体積を割ったもの

<Y> 要素数

(4) 要素高アスペクト比逆数分布

<X> 要素高アスペクト比の逆数 =  $\frac{2}{\sqrt{3}} \frac{\min(\text{要素の高さ})}{\max(\text{エッジ長})}$

<Y> 要素数

(5) 最小要素高さ分布

<X> 要素の最も低い高さ ( 実寸 )

<Y> 要素数

密度比分布 ( *option* )

<X> 要素の重心における節点間隔で最小要素高さを割ったもの

<Y> 要素数

## 5.2 プログラムのコマンドオプション

- p VRML ファイル出力オプション . 本オプションを指定すると , 指定した表面パッチデータファイル名に `_chk.wrl` を付加したものと `_har.wrl` を付加したものの 2 つの VRML ファイルが出力されます .
- ファイル名に `_chk.wrl` が付加された VRML ファイルは , 二面挟角最小値が基準値より小さな要素 , 二面挟角最大値が基準値より大きな要素 , また , 辺長換算体積が負であった要素が表示されます .
- ファイル名に `_har.wrl` が付加された VRML ファイルは , 要素高 ( 実寸 ) が最小の要素と要素高アスペクト比逆数が基準値より小さい要素が表示されます .
- d 密度比分布表示オプション . 節点密度データファイルを読み , 節点位置の節点間隔と , メッシュのエッジ長などとの比を出力します . なお , 節点密度データファイルの拡張子を除いたファイル名は , 四面体メッシュデータの拡張子を除いたものと同じである必要があります .

## 6. ファイル仕様

本プログラムで使用するファイルを以下に示します。

ファイル名	ファイルの概要
表面パッチデータファイル (.pcm)	節点座標, 三角形パッチの情報を含むデータファイル (頂点座標・三角形コネクティビティ)
表面メッシュデータファイル (.pcc)	TetMesh_P で作成された節点座標, 三角形表面メッシュの情報を含むデータファイル(中間ファイル) (頂点座標・三角形コネクティビティ・その他)
節点密度データファイル (.ptn)	節点疎密制御に使用するデータファイル
メッシュデータファイル (.msh)	本プログラムで出力される節点座標, 四面体メッシュの情報を含むデータファイル (節点座標・四面体コネクティビティ)
VRML ファイル (.vr1)	表面パッチ, 表面メッシュまたはメッシュ表面をVRML 形式に変換したファイル(VRML フォーマット Ver1.0)



## 6.1 表面パッチデータファイル

以下に表面パッチデータのフォーマットを示します。

- ・表面パッチの法線ベクトルは、形状の内部方向に向かうように設定されています(コネクティビティが形状の外から見て右回り)。
- ・頂点番号は0から始まります。
- ・拡張子は `pcm` です。

```

1629 0 2                ← 頂点数 予備0 ポリユーム(領域)数
150 -50 50             ← 1件目の頂点のX,Y,Z座標
50 -50 50
150 50 50
50 50 50
    ~ 省略 ~
50 17.03994 -22.52797
50 20.23377 -15.25734
50 29.21514 -26.66399
50 41.96536 -15.88812    ← 1629件目の頂点のX,Y,Z座標
1598 0 0               ← 第1ポリユームの表面パッチ数 予備0 予備0
158 128 17             ← 1件目の表面パッチを構成する頂点番号の並び
17 128 16
16 160 15
    ~ 省略 ~
738 704 799
794 800 731
800 778 731             ← 1598件目の表面パッチを構成する頂点番号の並び
1652 0 0               ← 第2ポリユームの表面パッチ数 予備0 予備0
960 958 1035           ← 1件目の表面パッチを構成する頂点番号の並び
841 1025 959
816 817 930
    ~ 省略 ~
1566 1627 1621
926 1614 1628
1586 1628 1615         ← 1652件目の表面パッチを構成する頂点番号の並び

```

注) 複数材料の場合、表面パッチは材料毎に閉じている必要があります。ここでは各材料をポリユームと呼んでいます。各頂点も要素もポリユーム毎にユニークであり、同じ頂点が複数のポリユームから参照されてはなりません。また、二つのポリユームが接する境界面では各頂点も各要素も空間的に一致している必要があります。従って、ポリユーム境界では同じ座標値を持った頂点が2つあり、それぞれが別のポリユームに属する要素から参照されていることとなります。各要素(三角形)もそれぞれ対となる(座標が一致している)頂点を結び、境界面上で二重になっている必要があります。ただし、TetMesh\_Mで最終的に生成される四面体メッシュでは、境界面上で共有節点になっています。ADVENTURE\_TriPatchのマニュアルも参照してください。

## 6.2 節点密度制御ファイル

### (1) 節点密度制御データの概要

節点密度データは、基本節点間隔とローカル節点密度に分類されます。

#### a. 基本節点間隔

メッシュの基本となる稜線長を指定します。この長さに従うようにメッシュが作成されます。

#### b. ローカル節点密度

入力形状における任意の箇所のメッシュを細かくしたい場合に利用します。ローカル節点密度は“点からの距離に反比例”、“線分からの距離に反比例”（2パターン）があります。ローカル節点密度を指定する場合、適用範囲、密度の強さのパラメータを設定します。

### (2) 節点密度適用例

図 6.2-1～図 6.2-3 に節点密度適用例（注）を示します。節点密度適用例は“点からの距離に反比例”、“線分からの距離に反比例”と“円筒からの距離に反比例（線分からの距離に反比例の一種）”の計 3 パターンについて示してあります。

- ・各パターンの左図は、適用結果図、右図は密度と距離の関係図を示しています。
- ・密度と距離の関係図の横軸  $r$  または  $r_1 \sim r_4$  は距離、縦軸  $d$  は密度です。
- ・密度と距離の関係図の距離とは、“点からの距離に反比例”の場合指定した点からの距離を示し、“線分からの距離に反比例”の場合指定した線分からの距離を示します。

<例>

例として図 6.2-1 “点からの距離に反比例”について説明します。この密度を適用すると、点からの距離が大きくなるに従って節点密度が低下していきます（点から離れると節点間隔が大きくなる）。

注) トップディレクトリの下に `sample_data` に節点密度制御のサンプルデータが格納されています (`adventure_manual_data02.ptn`) 。

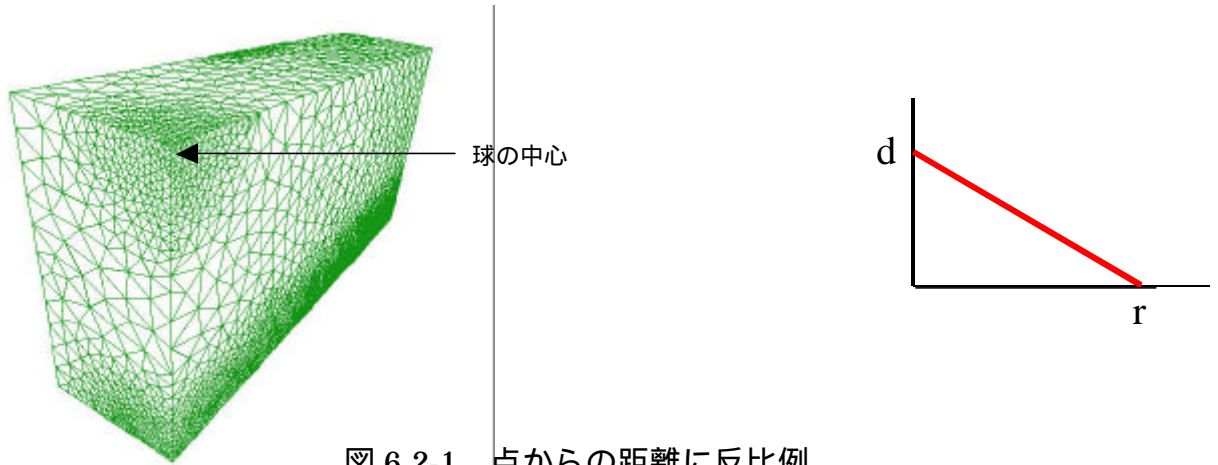


図 6.2-1 点からの距離に反比例  
( NodalPatternOnPoint を使用 )

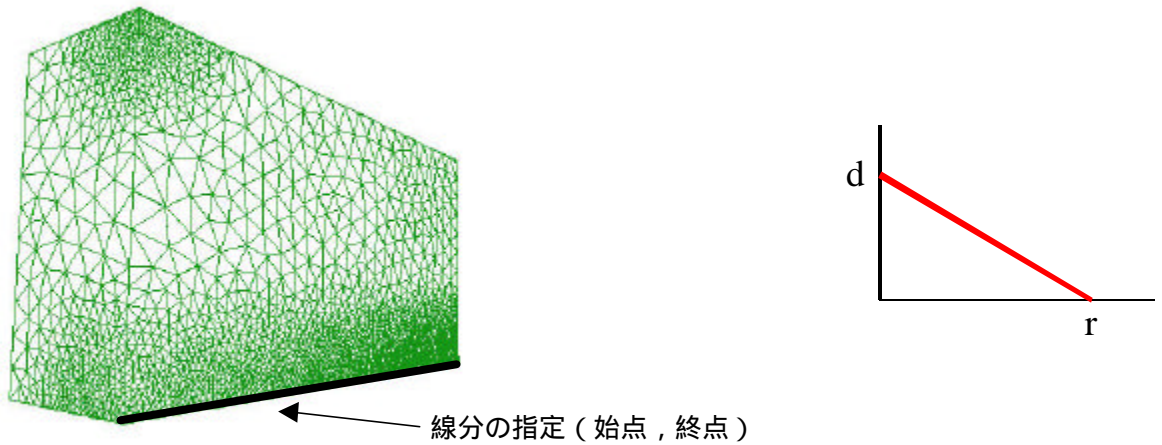


図 6.2-2 線分からの距離に反比例  
( NodalPatternOnLine を使用 )

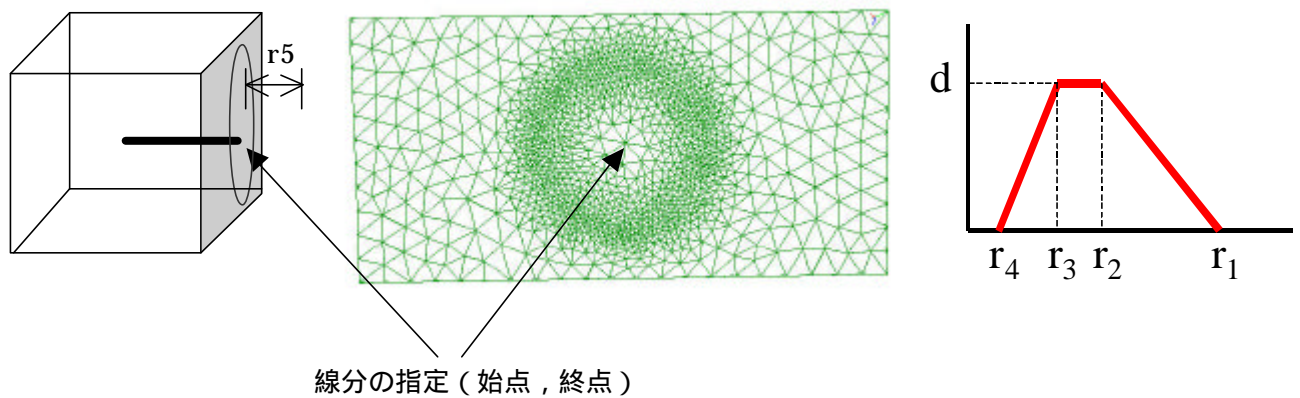


図 6.2-3 円筒からの距離に反比例  
( NodalPatternOnCylinder を使用 )

### (3) 節点密度制御ファイルのフォーマット

以下に節点密度制御データのフォーマットを示します。

- ・プログラムを実行する際、必須項目は `BaseDistance` です。
- ・その他の項目 (`NodalPatternOnPoint` , `NodalPatternOnLine` , `NodalPatternOnCylinder`) は入力形状における任意の箇所のメッシュを細かくしたい場合に利用します。
- ・拡張子は `ptn` です。

<code>BaseDistance</code>	← 基本節点間隔
5.0	
<code>NodalPatternOnPoint</code>	← 点からの距離に反比例
20 5.0	← 球の中心からの範囲 ( <b>r</b> ) 密度の強さ
0.0 50.0 25.0	← 球の中心座標
<code>NodalPatternOnLine</code>	← 線分からの距離に反比例
20 5.0	← 線分からの範囲 ( <b>r</b> ) 密度の強さ
0.0 0.0 20.0	← 線分の始点座標
100.0 0.0 25.0	← 線分の終点座標
<code>NodalPatternOnCylinder</code>	← 円筒からの距離に反比例 (節点密度の範囲指定が可能)
25.0 17.0 15.0 3.0 1.0 4.0	← 範囲1 ~ 範囲5 ( <b>r1 ~ r5</b> ) 密度の強さ
50.0 25.0 0.0	← 線分の始点座標
50.0 25.0 5.0	← 線分の終点座標

### 6.3 メッシュデータファイル

以下に四面体メッシュデータのフォーマットを示します。

- ・メッシュのコネクティビティについては 図 6.3-1 を参照してください。
- ・節点番号，要素番号は 0 から始まります。
- ・拡張子は `msh` です。

```

170776                ← 全要素数
19900 19890 22150 22160  ← 1 件目の要素を構成する節点並び
730 60 58 61
730 61 58 62
    ~ 省略 ~
38139 38601 38602 38606
38274 38139 38266 38607    ← 170776 件目の要素を構成する節点並び
38608                    ← 全節点数
-31.223900 -3.384220 -5.000000  ← 1 件目の節点の座標
-31.223900 -3.384220 -3.430000
    ~ 省略 ~
31.252200 3.060460 5.000000
31.223900 3.384220 5.000000    ← 38608 件目の節点の座標
2                                ← ボリューム (領域) 数
2567                            ← 第 1 ボリュームの要素数
0                                ← 第 1 ボリュームの 1 件目の要素番号
1
2
    ~ 省略 ~
2566                            ← 第 1 ボリュームの 2567 件目の要素番号
2052                            ← 第 2 ボリュームの要素数
2567                            ← 第 2 ボリュームの 1 件目の要素番号
2568
    ~ 省略 ~
4617
4618                            ← 第 2 ボリュームの 2052 件目の要素番号

```

注) 上記は四面体一次要素の場合。二次要素の場合には要素のコネクティビティが 10 個となります。

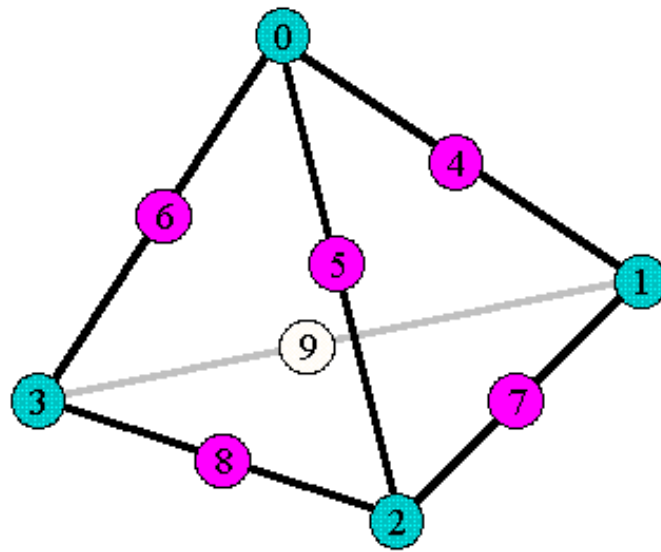


図 6.3-1 四面体メッシュの節点コネクティビティ

**参考文献**

- (1) Frank J. Bossen, Paul S. Heckbert, "A Pliant Method for Anisotropic Mesh Generation", 5<sup>th</sup> Annual International Meshing Roundtable, (1996).
- (2) Yagawa, G., Yoshimura, S. and Nakao, K., "Automatic Mesh Generation of Complex Geometries Based on Fuzzy Knowledge Processing and Computational Geometry", Integrated Computer-Aided Engineering 2, pp.265-282, (1995).

## 付録

## 付録 A TetMesh\_P の実行ログ (単一材料)

以下に、表面メッシュ生成プログラム TetMesh\_P の実行ログの説明を示します。

ADVENTURE TetMesh_P		プログラム名
input patch file:adventure_manual_data01.pcm		表面パッチファイル名
number of input vertices	= 2213	頂点数
number of volumes	= 1	ボリューム数
number of input elements	= 4422	三角形表面パッチ数
range of x-axis	= -7.1576560E+01 -2.1320670E+01	X 座標範囲
range of y-axis	= -1.6141980E+00 4.8337110E+01	Y 座標範囲
range of z-axis	= 0.0000000E+00 1.0000000E+01	Z 座標範囲
input density control file:adventure_manual_data01.ptn		節点密度制御ファイル名
BaseDistance	= 2.5000000E+00	基本節点間隔
number of density function	= 1	節点密度制御関数の数
maximum range	= 2.0000000E+01	最大影響範囲
maximum strength	= 3.5000000E+00	最大強度
number of edges	= 6633	入力表面パッチのエッジの情報
minimum edge length	= 4.1844367E-01	555 737
maximum edge length	= 3.5376171E+00	2 326
average edge length	= 1.3090199E+00	
Check Surfaces		
number of surface	= 1	面の数
Edge Correction start		
iteration loop, change count	= 1 0	削除したエッジの数
iteration loop, change count	= 2 0	
Surface Patch Grouping		表面のグループ化処理
number of Volumes	= 1	ボリューム (材料) 数
number of Bodies	= 1	ボディ数
number of Surfaces	= 1	サーフェス数
number of fixed main vertices	= 12	固定頂点数
number of boundary edge groups	= 18	境界エッジグループ数
open edge group	= 18	
closed edge group	= 0	
fixed edge	= 317	
number of face groups	= 8	面グループ数
Node bucket registration		節点をバケットに登録



Delaunay re-triangulation

Delaunay 再分割化

LEPP - Rough vertex density control start

形の悪い要素の分割  
(繰返し計算)

iteration loop, change count = 1 1  
iteration loop, change count = 2 1  
iteration loop, change count = 3 0

LEPP - Rough vertex density control : iteration converged 収束しなくても良い

Shape dependent density control .....

形状による節点密度制御関数の発生  
節点密度制御データの分布に従って  
節点を追加(繰返し計算)

Vertex density control start

iteration loop, change count = 1 2  
iteration loop, change count = 2 0

Vertex density control : iteration converged

収束しなくても良い

Pre-smoothing of boundary edge

境界エッジの点のみ先行平滑化

Pliant Delaunay retriangulation start

平滑化開始  
収束ループ

outer/inner iteration, remained = 1 1 1986  
outer/inner iteration, remained = 1 2 1997  
outer/inner iteration, remained = 1 3 1946  
outer/inner iteration, remained = 1 4 1855

~ 省略 ~

outer/inner iteration, remained = 1 198 9  
outer/inner iteration, remained = 1 199 12  
outer/inner iteration, remained = 1 200 10

\*\*\*\* inner iteration not converged \*\*\*\*

outer/inner iteration, remained = 2 1 7  
outer/inner iteration, remained = 2 2 2  
outer/inner iteration, remained = 2 3 1  
outer/inner iteration, remained = 2 4 0

\*\*\*\* inner iteration converged \*\*\*\*

--- outer iteration converged ----- loop 2

内側ループ収束  
外側ループ収束  
Laplacian smoothing 再平滑化

Laplacian smoothing start

outer/inner iteration, remained = 3 1 2173  
outer/inner iteration, remained = 3 2 251  
outer/inner iteration, remained = 3 3 161

~ 省略 ~

outer/inner iteration, remained = 3 8 3  
outer/inner iteration, remained = 3 9 1  
outer/inner iteration, remained = 3 10 0

\*\*\*\* inner iteration converged \*\*\*\*

内側ループ収束  
境界エッジ保護

boundary edge protection : outer loop = 3  
boundary edge protection : change count = 0

surface protection : outer loop = 3  
surface protection : change count = 0

表面保護

--- outer iteration converged ----- loop 3

外側ループ収束

number of vertices = 2185

出力頂点数

number of elements = 4366

出力要素数

## ADVENTURE SYSTEM

open:adventure\_manual\_data01c.pcc  
open:adventure\_manual\_data01c.ptn  
open:adventure\_manual\_data01\_c.wrl

出力表面メッシュ中間ファイル名  
出力節点密度制御ファイル名  
出力 VRML ファイル名

maximum allocate 3762383 Bytes 3.588 MBytes

start: Thu Mar 6 21:05:18 2003

stop : Thu Mar 6 21:05:30 2003

elapsed 9.13 sec (user)  
0.09 sec (system)  
9.22 sec (total)

## 付録 B TetMesh\_M の実行ログ (単一材料)

以下に，四面体メッシュ生成プログラム TetMesh\_M の実行ログの説明を示します．

ADVENTURE TetMesh_M	プログラム名
read densityFunction << adventure_manual_data01c.ptn	入力節点密度データ名
read file << adventure_manual_data01c.pcc	入力表面メッシュデータ名
read domain patch	領域内パッチ読み込み
total vertices = 2185	総節点数
total number of volume = 1	領域 (ボリウム) 数
set domain data	領域内のデータを設定
set interior nodes	
set local patches and vertices :: region number = 0	領域 0 パッチ・節点読み込み
local use total nodes = 2185	領域内でのパッチ・節点数
domain patches = 4366	
number of vertices = 2185	
number of patches = 4366	
set duplicate vertices = 0	二重節点数
bounding box ( -71.5766, -1.6142, 0 ) ( -21.3207, 48.3371, 10 )	座標範囲
baseMeshSize = 2.5	基本節点間隔
minInterval = 0.714286	最小節点間隔
node Generation on Vertex	形状表面節点発生
add vertices = 2185	追加表面節点数
add interior nodes = 0	追加 内部節点数
node Generation in Body	形状内部節点発生開始
bucket ----- 0/1520	節点発生終了バケット数
number of nodes 2185	累積節点数
bucket ----- 76/1520	
number of nodes 2185	
bucket ----- 152/1520	
number of nodes 2528	
bucket ----- 228/1520	
number of nodes 3064	
bucket ----- 304/1520	
number of nodes 3756	
bucket ----- 380/1520	
~ 省略 ~	
bucket ----- 1140/1520	
number of nodes 4976	
bucket ----- 1216/1520	
number of nodes 4983	
bucket ----- 1292/1520	
number of nodes 4989	
bucket ----- 1368/1520	
number of nodes 4996	
bucket ----- 1444/1520	
number of nodes 4996	
bucket ----- 1520/1520	
number of nodes 4996	バケット法による節点発生終了

```

Delaunay Triangulation
add Points
remove Outer Tetrahedron
correct Sliver Elements
number of additional points for sliver loop-1 = 275
total number of points = 5271
number of additional points for sliver loop-2 = 6
total number of points = 5277
-----count On Vertex = 2185
-----count In Body   = 3092
            total lry node = 5277
number of Elements      = 25812
write .wrl >> adventure_manual_data01c_e.wrl
write .wrl >> adventure_manual_data01c_n.wrl
clear all

total -----
number of total nodes      = 5277          総節点数
volume 0 = 5277 : 2185 ( v )      0 ( dv ) 3092 ( b )
            領域( ボリューム )番号 総節点数 表面節点数(他領域の節点との重複が無いもの)
            重複表面節点数(他領域の節点と重複しているもの) 内部節点数
number of total Elements  = 25812          総要素数
volume 0 = 25812          領域( ボリューム )0 要素数

write .msh >> adventure_manual_data01c.msh      出力メッシュファイル名

start : Thu Mar 6 21:07:19 2003
end   : Thu Mar 6 21:07:37 2003
interval = 18
process time = 14.78

END advtmesh9m

```

Delaunay 要素分割開始

外部要素削除後の要素数  
スライバー要素修正開始  
追加した節点数

表面節点数  
内部節点数  
節点数合計  
要素数

出力 VRML ファイル名 (要素表面)  
出力 VRML ファイル名 (節点)  
データ削除

総節点数

領域( ボリューム )番号 総節点数 表面節点数(他領域の節点との重複が無いもの)  
重複表面節点数(他領域の節点と重複しているもの) 内部節点数

総要素数

領域( ボリューム )0 要素数

出力メッシュファイル名

## 付録 C TetMesh\_S の実行ログ (単一材料)

以下に、二次節点追加プログラム TetMesh\_S の実行ログについて説明します。

reading... adventure_manual_data01c.msh	入力メッシュファイル名
linear tetrahedron ---> quadratic tetrahedron	
number of nodes = 5277	節点数
number of elements = 25812	要素数
writing... adventure_manual_data01cs.msh	出力メッシュファイル名
number of nodes = 38548	節点数
number of elements = 25812	要素数
number of edges = 33271	エッジ数
DOF(1ry) = 15831	自由度数 (一次)
DOF(2ry) = 115644	自由度数 (二次)
number of regions = 1	領域 (ボリューム) 数
range of x-axis = -7.157660e+01 -2.132070e+01	X 座標範囲
range of y-axis = -1.614200e+00 4.833710e+01	Y 座標範囲
range of z-axis = 0.000000e+00 1.000000e+01	Z 座標範囲
elapsed time = 1.07 sec	

## 付録 D TetMesh\_P の実行ログ (複数材料)

以下に、入力データが複数領域 (複数ボリューム) である場合の表面メッシュ生成プログラム TetMesh\_P の実行ログについて説明します。

ADVENTURE TetMesh_P		プログラム名
input patch file:mat_in0102.pcm		表面パッチファイル名
number of input vertices = 1629		頂点数
number of volumes = 2		ボリューム数
Volume 1		
number of input elements = 1598		
Volume 2		
number of input elements = 1652		
total number of input elements = 3250		三角形表面パッチ数
range of x-axis = -5.0000000E+01	1.5000000E+02	X 座標範囲
range of y-axis = -5.0000000E+01	5.0000000E+01	Y 座標範囲
range of z-axis = -5.0000000E+01	5.0000000E+01	Z 座標範囲
input density control file:mat_in0102.ptn		節点密度制御ファイル名
BaseDistance = 1.0000000E+01		基本節点間隔
number of density function = 0		節点密度制御関数の数
number of edges = 4875		入力表面パッチのエッジの情報
minimum edge length = 5.7324549E+00	694	770
maximum edge length = 1.5546799E+01	605	633
average edge length = 9.3825988E+00		
Check Surfaces		
Volume 1		
number of surface = 1		面の数
Volume 2		
number of surface = 1		
maximum number of dup.vertex = 1		最大重複節点定義数
maximum number of dup.edge = 1		最大重複エッジ定義数
Edge Correction start		極端につぶれた要素の削除
iteration loop, change count = 1 0		(繰返し計算)
iteration loop, change count = 2 0		削除したエッジの数
Surface Patch Grouping		表面のグループ化処理
number of Volumes = 2		ボリューム (材料) 数
number of Bodies = 2		ボディ数
number of Surfaces = 2		サーフェス数
number of fixed main vertices = 16		固定頂点数
number of boundary edge groups = 24		境界エッジグループ数
open edge group = 24		

```

closed edge group          =          0
fixed edge                 =         264
number of face groups     =          12
面グループ数

Node bucket registration   節点をバケットに登録

Delaunay re-triangulation Delaunay 再分割化

LEPP - Rough vertex density control start
iteration loop, change count =    1      0
LEPP - Rough vertex density control : iteration converged 収束しなくても良い

Shape dependent density control .....
Vertex density control start
iteration loop, change count =    1      0
Vertex density control : iteration converged
形状による節点密度制御関数の発生
節点密度制御データの分布に従って
節点を追加 (繰返し計算)
収束しなくても良い

Pre-smoothing of boundary edge 境界エッジの点のみ先行平滑化

Pliant Delaunay retriangulation start
outer/inner iteration, remained =    1    1    1422
outer/inner iteration, remained =    1    2    1424
outer/inner iteration, remained =    1    3    1346
平滑化開始
収束ループ
~ 省略 ~
outer/inner iteration, remained =    1   37     4
outer/inner iteration, remained =    1   38     2
outer/inner iteration, remained =    1   39     0
**** inner iteration converged ****
--- outer iteration converged ----- loop    1
Laplacian smoothing start
outer/inner iteration, remained =    2    1    1394
outer/inner iteration, remained =    2    2    110
outer/inner iteration, remained =    2    3     72
outer/inner iteration, remained =    2    4     36
outer/inner iteration, remained =    2    5     16
outer/inner iteration, remained =    2    6     4
outer/inner iteration, remained =    2    7     1
outer/inner iteration, remained =    2    8     0
Laplacian 再平滑化
**** inner iteration converged ****
boundary edge protection : outer loop =    2
境界エッジ保護
boundary edge protection : change count =    0
surface protection      : outer loop =    2
表面保護
surface protection      : change count =    0
duplicate edge protection : outer loop =    2
重複エッジ保護
duplicate edge protection : change count =    0
--- outer iteration converged ----- loop    2
外側ループ収束

number of vertices      =    1395
出力頂点数
number of elements      =    3088
出力要素数

open:mat_in0102c.pcc
出力表面メッシュ中間ファイル名

```

## ADVENTURE SYSTEM

open:mat\_in0102c.ptn  
open:mat\_in0102\_c.wrl

出力節点密度制御ファイル名  
出力 VRML ファイル名

maximum allocate 2801176 Bytes 2.671 MBytes

start: Thu Mar 6 21:06:28 2003  
stop : Thu Mar 6 21:06:34 2003

elapsed 5.40 sec (user)  
0.11 sec (system)  
5.51 sec (total)



## 付録 E TetMesh\_M の実行ログ (複数材料)

以下に、入力データが複数領域 (複数ボリューム) である場合の四面体メッシュ生成プログラム TetMesh\_M の実行ログについて説明します。

ADVENTURE TetMesh_M	プログラム名
read densityFunction << mat_in0102c.ptn	入力節点密度ファイル
read file << mat_in0102c.pcc	入力表面メッシュファイル
read domain patch	領域データ設定
total vertices = 1395	総節点数
total number of volume = 2	領域 (ボリューム) 数
set domain data	領域内データ設定
set interior nodes	
set local patches and vertices :: region number = 0	領域番号 0 のパッチ・節点の設定
local use total nodes = 1395	
domain patches = 1540	領域内パッチ
number of vertices = 772	領域内使用節点数
number of patches = 1540	領域内パッチ
set duplicate vertices = 0	2重節点数
bounding box ( 5, -5, -5 ) ( 15, 5, 5 )	座標範囲
baseMeshSize = 1	基本節点間隔
minInterval = 1	最小節点間隔
node Generation on Vertex	形状表面節点発生
add vertices = 772	追加表面節点数
add interior nodes = 0	追加内部節点数
node Generation in Body	形状内部節点発生開始
bucket ----- 0/64	節点発生終了パケット数
number of nodes 772	累積節点数
bucket ----- 16/64	
number of nodes 897	
bucket ----- 32/64	
number of nodes 1101	
bucket ----- 48/64	
number of nodes 1300	
bucket ----- 64/64	
number of nodes 1396	
Delaunay Triangulation	Delaunay 要素分割開始
add Points	
remove Outer Tetrahedron	外部要素削除
correct Sliver Elements	スライバー要素修正開始
number of additional points for sliver loop-1 = 59	追加した節点数
total number of points = 1455	
number of additional points for sliver loop-2 = 1	
total number of points = 1456	
-----count On Vertex = 772	領域 0 表面節点数
-----count In Body = 684	領域 0 内部節点数
total lry node = 1456	領域 0 節点数合計
number of Elements = 6491	領域 0 要素数合計

```

write .wrl >> mat_in0102c_e.wrl
write .wrl >> mat_in0102c_n.wrl
clear all
set domain data
set interior nodes
set local patches and vertices :: region number = 1
local use total nodes      = 1395
domain patches             = 1548
number of vertices         = 776
number of patches          = 1548
  set duplicate vertices   = 0
bounding box ( -5, -5, -5 ) ( 5, 5, 5 )
baseMeshSize = 1
minInterval  = 1
node Generation on Vertex
  add vertices      = 776
  add interior nodes = 0
node Generation in Body
bucket ----- 0/64
number of nodes 776
bucket ----- 16/64
number of nodes 901
bucket ----- 32/64
number of nodes 1107
bucket ----- 48/64
number of nodes 1309
bucket ----- 64/64
number of nodes 1402
Delaunay Triangulation
add Points
remove Outer Tetrahedron
correct Sliver Elements
number of additional points for sliver loop-1 = 62
total number of points = 1464
number of additional points for sliver loop-2 = 1
total number of points = 1465
-----count On Vertex = 776
-----count In Body    = 689
          total lry node = 1465
number of Elements      = 6516
write .wrl >> mat_in0102c_e.wrl
write .wrl >> mat_in0102c_n.wrl
clear all

total -----
number of total nodes      = 2768
volume 0 = 1456 : 772 ( v )      0 ( dv )      684 ( b )
volume 1 = 1465 : 623 ( v )     153 ( dv )     689 ( b )
          領域 ( ボリューム ) 番号 総節点数 表面節点数 ( 他領域の節点との重複が無いもの )
          重複表面節点数 ( 他領域の節点と重複しているもの ) 内部節点数
number of total Elements = 13007
          総要素数
          volume 0 = 6491
          領域 0 要素数

```

出力 VRML ファイル名 ( 要素表面 )

出力 VRML ファイル名 ( 節点 )

領域 0 データ削除

領域 1 内の要素作成開始

領域 1 パッチ・節点設定

( 以下領域 0 と同様 )

領域 1 表面節点数

領域 1 内部節点数

領域 1 総節点数

領域 1 要素数

総節点数

0 ( dv ) 684 ( b )

153 ( dv ) 689 ( b )

領域 ( ボリューム ) 番号 総節点数 表面節点数 ( 他領域の節点との重複が無いもの )

重複表面節点数 ( 他領域の節点と重複しているもの ) 内部節点数

総要素数

領域 0 要素数

volume 1 = 6516

領域 1 要素数

write .msh >> mat\_in0102c.msh

出力メッシュファイル名

start : Thu Mar 6 21:09:51 2003

end : Thu Mar 6 21:09:57 2003

interval = 6

process time = 5.52

END advtmesh9m

## 付録 F TetMesh\_S の実行ログ (複数材料)

以下に、入力データが複数領域 (複数ボリューム) である場合の二次節点追加プログラム TetMesh\_S の実行ログについて説明します。

```

reading... mat_in0102c.msh                入力メッシュファイル
linear tetrahedron ---> quadratic tetrahedron
number of nodes    =          2768        節点数
number of elements=          13007        要素数

writing... mat_in0102cs.msh              出力メッシュファイル名
number of nodes    =          19826        総節点数
number of elements=          13007        総要素数
number of edges    =          17058        総エッジ数
DOF(1ry)           =           8304        自由度数 (一次)
DOF(2ry)           =          59478        自由度数 (二次)
number of regions  =           2          領域 (ボリューム) 数
range of x-axis    = -5.000000e+01  1.500000e+02  X 座標範囲
range of y-axis    = -5.000000e+01  5.000000e+01  Y 座標範囲
range of z-axis    = -5.000000e+01  5.000000e+01  Z 座標範囲

elapsed time = 0.56 sec

```

## 付録 G TetMesh\_E の実行ログ

TetMesh\_E は、以下の 5 項目を評価します。

- (1) エッジ長さ
- (2) 二面挟角
- (3) 辺長換算体積
- (4) 要素高アスペクト比逆数
- (5) 最小要素高さ

以下に、実行ログの内容について説明します。

```

input mesh file : mati_in0102cs.msh
quadratic element          要素タイプ (一次 : linear 二次 : quadratic)
number of elements        =      13007      要素数
number of nodes           =      19826      節点数
number of volume          =           2      領域 (ボリューム) 数
    volume                1 =       6491      領域1の要素数
    volume                2 =       6516      領域2の要素数
X coordinates range       = -5.0000000e+01  1.5000000e+02      X座標範囲
Y coordinates range       = -5.0000000e+01  5.0000000e+01      Y座標範囲
Z coordinates range       = -5.0000000e+01  5.0000000e+01      Z座標範囲

read densityFunction : mati_in0102cs.ptn      節点密度ファイル読み込み
base mesh size           =  1.0000000e+01      基本節点密度数
number of 1ry nodes      =       2768      一次節点数
number of 2ry nodes      =       17058      二次節点数
number of total triangles =       27298      メッシュの総三角形数
number of surface triangles =       2568      メッシュ表面の三角形数
number of illegal elements =           0      不正な要素数
number of surface nodes  =       5138      表面節点数
    1ry surface nodes    =       1286      一次表面節点数
    2ry surface nodes    =       3852      二次表面節点数

number of total edges    =       17058      メッシュ総エッジ数
number of interior edges =       13206      内部エッジ数
number of surface edges  =       3852      表面エッジ数

Edge Length ----- ratio to local size --      エッジ長さ分布 密度比分布
 1  0.01      0  0.01      0      番号, エッジ長さ エッジ数 密度比 エッジ数
 2  0.10      0  0.10      0      エッジ長が0.01以上0.1未満のエッジ数
 3  0.20      0  0.20      0      密度比0.01以上0.1未満のエッジ数
 ~ 省略 ~
20 10.00     2028  1.90     77
21 20.00    13429  2.00     22
22 30.00      1   3.00      1

```

minimum edge length	=	4.7845943	最短エッジ長さ
maximum edge length	=	20.4460551	最長エッジ長さ
average edge length	=	11.7784436	平均エッジ長さ
minimum edge length ratio to local size	=	0.4784594	最小密度比
maximum edge length ratio to local size	=	2.0446055	最大密度比
average edge length ratio to local size	=	1.1778444	平均密度比

All of 2ry nodes are on the edge middle point.      二次節点がエッジの中点かチェック

Dihedral angle -- minimum	maximum	二面挟角分布	
1     5            0	0	番号	角度
2     10           1	0	最小角度要素数	最大角度要素数
3     15           63	0	要素の最小二面挟角が5度以上10度未満の要素数	
4     20           155	0		

~ 省略 ~

32    160           0	73		
33    165           0	4		
34    170           0	0		
35    175           0	0		
36    180           0	0	要素の最大二面挟角が175度以上180度以下の要素数	

minimum in minimum dihedral angle	=	9.2923577	最小最小二面挟角
maximum in minimum dihedral angle	=	70.1424314	最大最小二面挟角
minimum in maximum dihedral angle	=	71.9159013	最小最大二面挟角
maximum in maximum dihedral angle	=	164.2051316	最大最大二面挟角
average of minimum dihedral angle	=	46.8098209	平均最小二面挟角
average of maximum dihedral angle	=	102.6634915	平均最大二面挟角
average of dihedral angle	=	69.5994465	平均二面挟角
number of sliver elements	=	0	スライバー要素数
max > 175 & min < 5	=	0	最大最小の基準値外の要素数
max > 175	=	0	基準値(175度)より大きい要素数
min < 5	=	0	基準値(5度)より小さい要素数

Regular tetrahedral edge length	辺長換算体積 密度比分布		
distribution of equivalent element volume --	ratio to local size -		
1    0.01        0	0.01	0	番号 辺長 要素数
2    0.10        0	0.10	0	密度比 要素数
3    0.20        0	0.20	0	辺長が0.1以上0.2未満 および
4    0.30        0	0.30	0	密度比が0.1以上0.2未満の要素数
5    0.40        0	0.40	0	
6    0.50        0	0.50	0	
7    0.60        0	0.60	0	
8    0.70        0	0.70	14	
9    0.80        0	0.80	193	
10   0.90        0	0.90	645	
11   1.00        0	1.00	2009	
12   2.00        0	1.10	4362	
13   3.00        0	1.20	4028	
14   4.00        0	1.30	1386	
15   5.00        0	1.40	312	
16   6.00        0	1.50	50	
17   7.00        14	1.60	8	
18   8.00        193	1.70	0	

# ADVENTURE SYSTEM

19	9.00	645	1.80	0
20	10.00	2009	1.90	0
21	20.00	10146	2.00	0
22	30.00	0	3.00	0

minimum edge length	=	6.2320287	最小辺長換算体積
maximum edge length	=	15.8144499	最大辺長換算体積
average edge length	=	10.8046704	平均辺長換算体積
minimum edge length ratio to local size	=	0.6232029	最小辺長換算体積 密度比
maximum edge length ratio to local size	=	1.5814450	最大辺長換算体積 密度比
average edge length ratio to local size	=	1.0804670	平均辺長換算体積 密度比
number of illegal elements	=	0	不正要素数

( 体積が1.0e-10より小さい要素数 )

Inverse of Element Height Aspect Ratio --	要素高アスペクト比逆数分布		
1 0.05 0	番号	要素高アスペクト比逆数	要素数
2 0.10 0	要素高アスペクト比逆数が0.05以上0.1未満の要素数		
3 0.15 1			
4 0.20 42			
5 0.25 85			

~ 省略 ~

13	0.65	2017
14	0.70	1669
15	0.75	1002
16	0.80	548
17	0.85	246
18	0.90	138
19	0.95	30
20	1.00	7

minimum inv. element height aspect ratio	=	0.1266485	最小要素高アスペクト比逆数
maximum inv. element height aspect ratio	=	0.9763302	最大要素高アスペクト比逆数
average inv. element height aspect ratio	=	0.5775726	平均要素高アスペクト比逆数
number of lower than regulation( 0.050 )	=	0	基準値 ( 0.05 ) より 小さい要素数

Minimum Element Height -- ratio to local size -	最小要素高さ分布 密度比分布			
1 0.01 0 0.01 0	番号	最小要素高	要素数	密度比 要素数
2 0.10 0 0.10 0	最小要素高が0.01以上0.1未満の要素数および 密度比が0.01以上0.1未満の要素数			
3 0.20 0 0.20 0				
4 0.30 0 0.30 23				
5 0.40 0 0.40 124				

~ 省略 ~

18	8.00	4522	1.70	0
19	9.00	1923	1.80	0
20	10.00	193	1.90	0
21	20.00	33	2.00	0

minimum element height	=	1.6518785	最小要素高
element number	=	675	最小要素高の要素番号

## ADVENTURE SYSTEM

inv. element height aspect ratio	=	0.1266485	要素高アスペクト比逆数
minimum dihedral angle	=	9.2923577	最小二面挟角
maximum dihedral angle	=	164.2051316	最大二面挟角
maximum element height	=	11.6131432	最大要素高
average element height	=	6.8353189	平均要素高
minimum element height ratio to local size=	=	0.2023130	最小要素高密度比
maximum element height ratio to local size=	=	1.4223138	最大要素高密度比
average element height ratio to local size=	=	0.8371522	平均要素高密度比

write chk.wrl >> mati\_in0102cs\_chk.wrl  
write har.wrl >> mati\_in0102cs\_har.wrl

不正または質の悪い要素のVRML出力  
(オプション-p指定時)

elapsed time = 2.050 sec  
END advtmesh9e