

# 曲面モデルに基づくメッシュレス解析のための節点の生成

## Node Generation for Meshless Analysis based on Implicit Surface Modeling

長谷川恭子<sup>1)</sup>, 仲田晋<sup>2)</sup>, 田中覚<sup>3)</sup>

Kyoko HASEGAWA, Susumu NAKATA and Satoshi TANAKA

- 1) 立命館大学 COE 推進機構 (〒 525-8577 滋賀県草津市野路東 1-1-1, E-mail:hasegawa@cg.is.ritsumei.ac.jp)  
 2) 立命館大学情報理工学部 (〒 525-8577 滋賀県草津市野路東 1-1-1, E-mail:snakata@media.ritsumei.ac.jp)  
 3) 立命館大学情報理工学部 (〒 525-8577 滋賀県草津市野路東 1-1-1, E-mail:stanaka@media.ritsumei.ac.jp)

In this study, we present a method to generate adaptive nodes for three-dimensional (3D) meshless analysis. 3D models of the structural analysis are usually constructed as a combination of simple shape models generated using modeling tools such as CAD systems. On the other hand, in the field of computer graphics, there have been proposed a number of techniques that make it possible to create complex shape models from a set of point obtained by 3D range scanners. One effective way of the modeling is to represent the surface as an implicit function and the node generation method presented in this paper is based on the structure of the implicit function representation. We also present some numerical results of a 3D problem.

**Key Words:** PDE solver, Meshless Method, Implicit Surface, Node Generation

### 1. はじめに

一般に偏微分方程式の数値解析手法として有限要素法が広く用いられている。有限要素法では、対象物の要素分割や節点同士の接続情報が必要であり、特に3次元形状ではその接続情報の生成に多大な困難を伴うことがある。これに対して、メッシュレス法を用いた数値解析手法では、その接続情報を必要とせず要素分割の手間を省くことが可能である。メッシュレス法を用いた解析手法としては Element-Free Galerkin Method (EFGM)<sup>(1)</sup> や Meshless Local Petrov-Galerkin Method (MLPGM)<sup>(2)</sup> が代表的である。

EFGM は Moving Least Squares Approximation<sup>(3)</sup> を用いた形状関数の生成が特徴であり、そのため境界条件には特殊な取り扱いが必要となる。これに対し、境界条件を従来の有限要素法と同様に扱える手法として Radial Point Interpolation Method (RPIM)<sup>(4)</sup> が提案され、さらに3次元弾性問題へ適用されている<sup>(5)</sup>。

以上のように多様な研究が進められ、さまざまな応用が考えられてきた。一方、その対象物は CAD を用いてモデリングされることが多く、複雑な形状のモデリングは労力を要する作業になる。

これに対し、近年における形状計測装置の発達により、複雑な形状をもつ物体表面の点群データを容易に取得できるようになった。さらに、その点群データを用いた曲面モデルの生成技術が急速に発展している。特に、物体表面の点群からその形状を陰関数曲面としてモデリングする手法の研究が

盛んに行われてきた<sup>(6,7)</sup>。特に、Multi-level Partition of Unity (MPU) 法<sup>(6)</sup> は点群データが大規模である場合にも高速に陰関数を生成できるという特徴をもつ。

本研究では形状計測装置の実測値から生成される曲面モデルを解析の対象とする。このモデルは形状が複雑であるため、要素生成を必要としない解析手法であるメッシュレス法が適していると考えられる。

本研究の目的は、形状計測装置から得られた点群に対するモデリングを想定した RPIM を用いた解析に適した節点を生成することである。この節点の生成には、MPU 法を用いて形状を表現した際の関数の構造を利用する。

### 2. 解析対象の形状の表現と節点の生成

本研究では、対象物の形状は陰関数曲面として表現されるものとする。すなわち、形状は関数  $f: \mathbb{R}^3 \rightarrow \mathbb{R}$  に対し、 $f(x) = 0$  を満たす点  $x \in \mathbb{R}^3$  の集合として定義される。なお、曲面内部で  $f(x) > 0$  と仮定する。陰関数曲面モデルの生成手法として Turk らの手法<sup>(7)</sup> や MPU 法<sup>(6)</sup> などがあるが、本研究では大量点群を扱うため MPU 法を採用する。

曲面を構成するための点群を  $x_i (i = 1, 2, \dots, N)$ 、各点における曲面の法線を  $n_i (i = 1, 2, \dots, N)$  とし、全点群を含む矩形領域を  $\Omega$  とする。MPU 法では領域  $\Omega$  を再帰的に分割することで小領域  $\Omega_i (i = 1, 2, \dots, M)$  が階層的に構成される。Fig.1 に MPU 法における小領域の構成の概念図を示す。同図では、各小領域において色が濃くなるに伴い、階層が深くなるこ

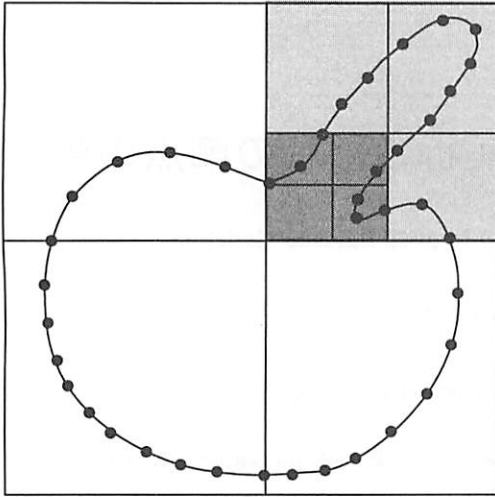


Fig. 1 Schematic view of reconstructing an implicit surface by MPU.

とを示している。ただし、同図は2次元における小領域の構成図であり、3次元の場合では小領域が再帰的に8分割される。以降、この小領域をセルと呼び、各セルの中心を  $c_i$  で表す。このとき、各セルで定義される関数  $\psi_i(\mathbf{x}), Q_i(\mathbf{x})$  を用いて  $f(\mathbf{x})$  を次のように表現する。

$$f(\mathbf{x}) = \sum_{i=1}^M \psi_i(\mathbf{x}) Q_i(\mathbf{x}), \quad \mathbf{x} \in \Omega. \quad (1)$$

ただし、 $\psi_i(\mathbf{x})$  は  $c_i$  を中心とする半径  $R_i$  の球をサポートにもつ重み関数であり、 $Q_i(\mathbf{x})$  はセル  $\Omega_i$  における曲面の形状を表す2次関数である。なお、各セルで定義される2次関数  $Q_i(\mathbf{x})$  は最小2乗法で近似し、以下の近似精度を満たすまで再帰的に分割する。

$$\max_{\|\mathbf{x}-c_i\|<R_i} \frac{|Q_i(\mathbf{x})|}{\|\nabla Q_i(\mathbf{x})\|} \leq \varepsilon_{\text{MPU}}. \quad (2)$$

この判定方法に従えば、セルは曲面付近や形状が複雑な部分では階層が深くその大きさも小さくなり、それ以外の部分では階層が浅く大きなセルが構成される。以上のようにMPU法では、関数  $f(\mathbf{x})$  が適応的に分割されたセル構造を用いて表現される。Fig.2には同手法を用いて生成した形状モデルを示す。ここで、 $\varepsilon_{\text{MPU}} = 10^{-4}$  である。

次に、形状計測装置により得られた点群は3次元形状の表面にのみ存在するため、形状の内部に解析のための節点を生成する必要がある。MPU法による陰関数曲面生成手法によれば、セルは階層的に生成される。また、陰関数曲面の微細構造を含む部分には小さなセルが多く生成され、曲面内部では大きなセルが生成される傾向がある。したがって、各セルの中心に節点を生成すれば、形状に依存した疎密をもつ節点を生成することができる。さらに、陰関数  $f(\mathbf{x})$  の値は形状の内部で正となりかつ外部で負となるため、生成した節点の内外判定が非常に容易であり、 $f(\mathbf{x}) > 0$  となる点群のみを生成すれば陰関数曲面内部に点群を生成できる。ただし、MPU

Table 1 Number of nodes in case of each level.

Level	Number of nodes
4	386
5	2329
6	7363
7	10939
8	11815
9	27519

法で生成されるセルの数は形状の複雑さに伴い増大し、Fig.2の場合においても全セル数は51885であり、最も浅い階層及び最も深い階層は4及び9である。従って、節点を生成する場合にはセルの階層を最下層まで見ず、途中の階層で打ち切ることとする。Table 1に各階層で打ち切った場合の節点数を示す。打ち切る階層を6として生成した節点をFig.3に示す。ただし、同図は  $0.3 \leq z \leq 0.6$  の節点のみを表示している。同図から曲面近傍で節点が密になっていることがわかる。

### 3. RPIMによる3次元Poisson問題の解法

複雑な形状に対して、陰関数曲面モデルの生成と形状の複雑さに依存した陰関数曲面内部の節点が生成できた、本節では、前節で生成したモデルと節点を適用した3次元Poisson問題の数値解法を考える。数値解法としてはRadial Point Interpolation Method (RPIM)<sup>(4)</sup>を採用する。

#### 3.1. RPIMにおける形状関数の構成

解析のための領域  $V \subset \mathbb{R}^3$  および節点  $\mathbf{x}_i \in V (i = 1, 2, \dots, n_{\text{total}})$  が与えられているものとする。このとき、RPIMでは、ある関数  $u_A(\mathbf{x})$  の近似関数  $u(\mathbf{x})$  は形状関数  $\varphi_i(\mathbf{x})$  および節点  $\mathbf{x}_i$  における  $u_A(\mathbf{x})$  の値  $u_i$  により、

$$u(\mathbf{x}) = [b^T(\mathbf{x}), p^T(\mathbf{x})] G^{-1} \begin{bmatrix} \mathbf{u} \\ 0 \end{bmatrix} = \varphi(\mathbf{x}) \mathbf{u}, \quad (3)$$

で定義される<sup>(5)</sup>。ただし、 $b(\mathbf{x}), p(\mathbf{x}), \varphi(\mathbf{x})$  はそれぞれ Radial basis function (RBF)  $b_i(\mathbf{x}) (i = 1, 2, \dots, n)$ , Polynomial basis function  $p_j(\mathbf{x}) (j = 1, 2, \dots, m)$ , 形状関数  $\varphi_i(\mathbf{x}) (i = 1, 2, \dots, n)$  を成分にもつベクトルであり、 $\mathbf{u}$  は  $u_A(\mathbf{x}_i)$  を成分にもつベクトルである。ここで、 $n$  は  $\mathbf{x}$  を中心とする局所的な球領域に含まれる節点の数であり、 $m$  は  $p(\mathbf{x})$  の次数である。また、上式中の  $G$  は

$$G = \begin{bmatrix} B_0 & P_0 \\ P_0^T & O \end{bmatrix}, \quad (4)$$

で定義される。ただし、(4)中の  $B_0, P_0$  はそれぞれ

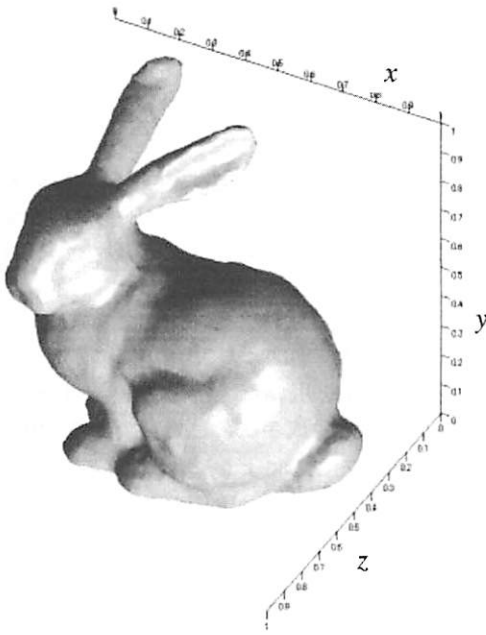


Fig.2 The implicit surface model reconstructed by MPU (Input data : Stanford 3D Scanning Repository).

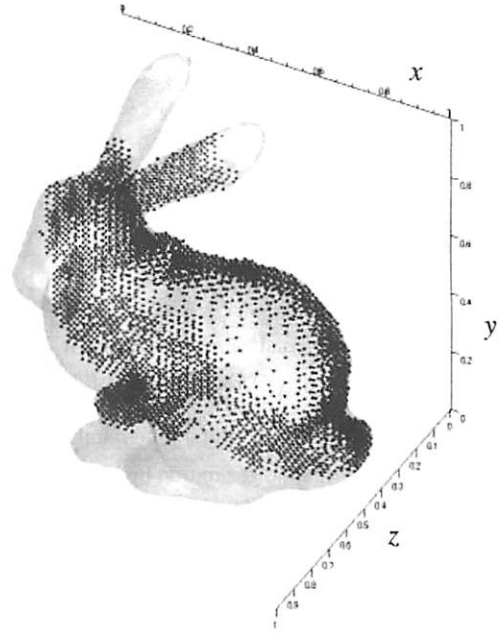


Fig.3 Nodes created inside the object.

$$B_0 = \begin{bmatrix} b_1(\mathbf{x}_1) & b_2(\mathbf{x}_1) & \cdots & b_n(\mathbf{x}_1) \\ b_1(\mathbf{x}_2) & b_2(\mathbf{x}_2) & \cdots & b_n(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ b_1(\mathbf{x}_n) & b_2(\mathbf{x}_n) & \cdots & b_n(\mathbf{x}_n) \end{bmatrix}, \quad (5)$$

$$P_0 = \begin{bmatrix} p_1(\mathbf{x}_1) & p_2(\mathbf{x}_1) & \cdots & p_m(\mathbf{x}_1) \\ p_1(\mathbf{x}_2) & p_2(\mathbf{x}_2) & \cdots & p_m(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ p_1(\mathbf{x}_n) & p_2(\mathbf{x}_n) & \cdots & p_m(\mathbf{x}_n) \end{bmatrix}, \quad (6)$$

で定義される. RPIM を用いて得られる形状関数  $\varphi_i(\mathbf{x})$  は  $\varphi_i(\mathbf{x}_j) = \delta_{ij}$  という性質をもつ. ここで,  $\delta_{ij}$  はクロネッカーのデルタを表す. この性質により, Dirichlet 境界条件は有限要素法と同様に容易に付加することができる<sup>(4)</sup>. 本研究では,  $b_i(\mathbf{x})$  として RBF<sup>(5)</sup> :

$$b_i(\mathbf{x}) = \begin{cases} \exp\left(-\alpha \frac{r_i^2}{c^2}\right) & r_i \leq R, \\ 0 & r_i > R, \end{cases} \quad (7)$$

を適用する. ただし,  $\alpha, c$  は定数であり,  $R$  は  $\mathbf{x}$  を中心とする球の半径である. また,  $r_i$  は  $r_i = \|\mathbf{x} - \mathbf{x}_i\|$  と定義する.

### 3.2. 3次元 Poisson 問題への適用

本研究では 3次元 Poisson 問題:

$$-\Delta u = p \quad \text{in } V, \quad (8)$$

$$u = \bar{u} \quad \text{on } \partial V, \quad (9)$$

を対象とする. ただし,  $V$  は曲面  $\partial V$  に囲まれる領域を示す.

また,  $\bar{u}$  は  $\partial V$  上で定義された既知関数である. ここで, (8) の弱形式は

$$\int_V \nabla u \cdot \nabla \delta u dV - \int_{\partial V} p \delta u dS = 0 \quad \text{for } \forall \delta u, \quad (10)$$

で書き表される.

式 (10) の弱形式を離散化する. RPIM では, 形状関数  $\varphi_i(\mathbf{x})$  を用いて  $u, \delta u$  の近似関数を

$$u(\mathbf{x}) = \sum_{i=1}^n u_i \varphi_i(\mathbf{x}), \quad (11)$$

$$\delta u(\mathbf{x}) = \sum_{i=1}^n \delta u_i \varphi_i(\mathbf{x}), \quad (12)$$

で書き表すことができる. 上式を用いて弱形式 (10) を離散化すれば

$$A\mathbf{x} = \mathbf{f}, \quad (13)$$

を得る. ただし,  $A, \mathbf{f}$  はそれぞれ

$$(A)_{ij} = \int_V \nabla \varphi_i \cdot \nabla \varphi_j dV, \quad (14)$$

$$(\mathbf{f})_i = \int_{\partial V} p \varphi_i dS, \quad (15)$$

であり,  $\mathbf{x}$  は未知数  $u_i$  を成分にもつベクトルである. (13) に対して, 有限要素法と同様の方法で Dirichlet 境界条件 (9) を付加することができ, このときの連立 1 次方程式を

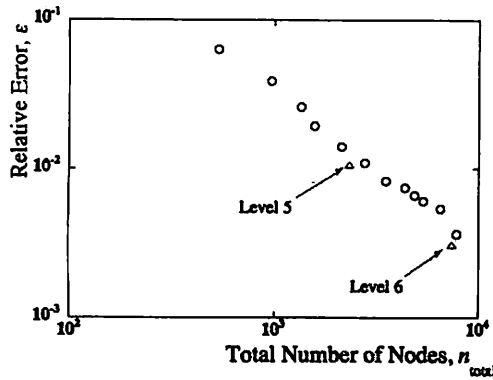


Fig.4 The relative error as a function of a total number of nodes.  
 ○:Node generation on grid, △:Node generation by MPU.

$$A^*x = f^*, \quad (16)$$

で書き表す。以上より、(16)を解くことで3次元 Poisson 問題の数値解を得ることができる。

#### 4. 数値例

本節では、Fig.2 に示す形状に対する 3 次元 Poisson 問題の数値解を求める。本研究では、Poisson 問題の解を  $u = \sin \pi x \cos \pi y \sin \pi z$  と想定し、これに基づいて Dirichlet 境界条件  $\bar{u}$  を与えている。また、RPIM の形状関数生成のためのパラメタとして、 $p(x)$  の次数  $m$  を 4 とし、係数  $\alpha$  は  $\alpha = 3.0, c$  は節点間の距離の平均値とする<sup>(4,5,8)</sup>。

精度の尺度として、相対誤差：

$$\varepsilon = \frac{\|u_A - u_N\|_\infty}{\|u_A\|_\infty}, \quad (17)$$

を採用する。ただし、 $u_A$  および  $u_N$  はそれぞれ解析解と数値解を表す。Fig.4 に相対誤差の節点数依存性を示す。ただし、同図中の○は格子上に節点を分布させ、陰関数曲面内部のみを取り出した節点を採用した場合における相対誤差を表す。同図に示す通り、節点数の増加に伴い相対誤差が減少していることがわかる。一方、△に MPU 法のセルの構造を適用して生成した節点を用いた場合を示す。セルの階層を 5 階層で打ち切って節点を生成した場合、曲面内部の節点数は 2329

であり、同図に示す通り、格子上に節点を生成した場合と同程度の精度の数値解が求められていることがわかる。セルの階層が 6 階層までの場合においても同様に節点が 7363 点生成され、格子上に生成した節点を用いた場合と同程度の結果が得られている。以上より、MPU 法のセル構造を適用して生成された節点を用いた場合でも、格子上に節点を生成した場合と同程度の精度の数値解が求められることがわかった。

#### 5. おわりに

本研究では、陰関数曲面としてモデリングされた複雑形状を対象とし、RPIM を用いた 3 次元 Poisson 問題の解析を実現した。曲面モデル生成手法のひとつである MPU 法の階層的な構造を利用することにより、形状に依存した疎密をもつ曲面内部の節点を生成することができる。

#### 参考文献

- (1) T. Belytschko, Y. Y. Lu and L. Gu : Element-free Galerkin methods, *Int. J. Numer. Methods Eng.*, **37** (1994), pp. 229–256.
- (2) S. N. Atluri and T. Zhu : A new meshless local Petrov-Galerkin (MLPG) approach in computational mechanics, *Comput. Mech.*, **22** (1998), pp. 117–127.
- (3) P. Lancaster and K. Salkauskas : Surface generated by moving least square methods, *Mathematics of Computation*, **37**-155 (1981), pp. 141–158.
- (4) J. G. Wang and G. R. Liu : A point interpolation meshless method based on radial basis function, *Int. J. Numer. Meth. Engng.*, **54** (2002), pp. 1623–1648.
- (5) G. R. Liu, G. Y. Zhang, Y. T. Gu and Y. Y. Wang : A meshfree radial point interpolation method (RPIM) for three-dimensional solids, *Comput. Mech.*, **36** (2005), pp. 421–430.
- (6) Y. Ohtake, A. Belyaev, M. Alexa, G. Turk and H. -P Seidel : Multi-level partition of unity implicits, *ACM Transactions on Graphics*, **22**-3 (2003), pp. 463–470.
- (7) G. Turk and J. F. O'Brien : Modeling with implicit surfaces that interpolate, *ACM Transactions on Graphics*, **21**-4 (2002), pp. 855–873.
- (8) G. R. Liu : Mesh free methods, (2003), CRC Press LLC.