

滑らかな境界をもつ領域のための等方的メッシュの生成法

56214 長井 超慧

指導教員 杉原 厚吉 教授

平成 18 年 2 月 5 日

1 概要

メッシュは流体力学やコンピュータグラフィックスなど多くの分野で用いられるデータ表現法である。2 次元三角形メッシュおよび 3 次元四面体メッシュに関して、滑らかな境界をもつ領域に対する等方的なメッシュ生成法の研究を行った。

2 次元三角形メッシュ: 等方性の向上を目的とする。Persson and Strang による、領域境界の曲線部分の表現に優れた等方的メッシュ生成アルゴリズム DistMesh を改良した。DistMesh で生成したメッシュの頂点密度が低い部分に頂点を追加し、スムージングを施してメッシュを再構成する手法を提案した。

3 次元四面体メッシュ: 等方性を保ちつつ、領域境界を忠実に表現することが目的である。四面体メッシュ表面上の頂点位置の決定に quadric error metric を用いて幾何誤差を減少した。頂点移動で等方性が低下するのを抑えるため、接続する要素の等方性を基に各頂点の移動距離を定めた。

2 2 次元三角形メッシュの等方性向上法の提案

本提案手法は Persson and Strang [3] による 2 次元三角形メッシュ生成アルゴリズム DistMesh の改良である。DistMesh で生成したメッシュに頂点を追加して、より等方的な 2 次元三角形メッシュを生成する。

2.1 DistMesh

生成領域の境界の形状を陰関数で表現することで、境界の曲線部分を滑らかに表現でき、境界上の頂点位置決定も容易になる。頂点同士の接続関係は Delaunay 三角形分割で定める。接続関係を固定したときの最適な頂点位置は、以下のエネルギー関数を最小化する位置とする。

$$\sum_{v_i \in V} \sum_{e_j \in E(v_i)} kq(l_j - l_j^*) \quad (1)$$

V はメッシュの頂点集合、 $E(v_i)$ は頂点 v_i に接続する辺集合、 k は定数、 l_j は辺 e_j の長さ、 l_j^* は辺 e_j の理想長を表す。また、関数 q は以下で定義される関数である:

$$q(x) = \begin{cases} x & (x > 0), \\ 0 & (x \leq 0). \end{cases} \quad (2)$$

辺の理想長は通常ユーザ指定の辺長関数で与えられる。しかし DistMesh では、ユーザは辺長変化を定める関数 h を入力し、それを基に独自の辺長関数:

$$h(x_{e_i}) \sqrt{\sum_j l_j^2 / \sum_j h^2(x_{e_j})} \quad (3)$$

を用いて理想長を定める (辺 e_i の指定は中点 x_{e_i} を与えることで行う)。このため相対的な辺長関数といえることができる。

頂点の初期位置はメッシュ生成領域内部の格子点である。また、頂点数が初期配置後に増減することはない。

2.2 頂点追加

DistMesh で生成した等方的なメッシュに対し、頂点を追加する以下のアルゴリズムを提案する。

アルゴリズム 頂点追加

入力: DistMesh による等方的な 2 次元三角形メッシュ、このメッシュを生成する際 DistMesh に与えた入力

出力: 等方的な 2 次元三角形メッシュ

1. 頂点密度判定: メッシュ内部の頂点 v_i を中心とし、 v_i における辺長関数値を半径とする円 c を考える。(v_i

の隣接頂点同士の間の辺長の総和) - (c の周長) の値が閾値より大きいなら、 v_i 周辺は密度が低いとする。以上の判定をメッシュ内部の全頂点に対して行う。頂点密度が低いと判断された頂点がなければ 4. へ。

2. 頂点の追加: 1. で周囲の頂点密度が低いと判断された頂点 v_i の近傍に新たに頂点を 1 つ追加する。 v_i に接続する辺のうち、 l_j/l_j^* が最大の辺 e_j を特定する。この辺に接続する 2 つの三角形のうち、外接円の半径が大きい方の外心を、頂点追加位置とする。

3. 接続関係と頂点位置の更新: DistMesh に従って全頂点を移動し、Delaunay 三角形分割を行うことを収束するまで繰り返す。1. へ。

4. スムージング: 一様メッシュにはラプラシアンスムージングを、非一様メッシュには optimal Delaunay triangulation スムージング [2] を施す。

DistMesh の辺長関数は辺長の変化を用いて辺長を定めるため、アルゴリズムの反復のたびに異なる関数値を返す。従ってこのままだと、アルゴリズム 頂点追加 1. の判定基準も反復のたびに変わってしまい、アルゴリズムが収束しないことがある。これを防ぐため、本手法では新たに絶対的な辺長関数:

$$h(x_{e_i}) \sqrt{\sum_j l_j^2 / H} \quad (4)$$

を定義し、頂点密度判定の際にはこれを用いる。 H は入力メッシュの $\sum_j h^2(x_{e_j})$ の値である。

頂点 v_i の次数を n_i 、隣接する頂点の座標を $v_{i,j}$ とする (以下本稿中では、頂点 v の座標を \mathbf{v} で表記する)。頂点 v_i の移動先はラプラシアンスムージングでは

$$\frac{1}{n_i} \sum_{j=1}^{n_i} v_{i,j} \quad (5)$$

となる。また、 $|T_{i,j}|$ で v_i に接続する要素 (2 次元では三角形) $T_{i,j}$ の体積 (2 次元では面積) を、 $\mathbf{c}_{i,j}$ で要素 $T_{i,j}$ の外心の座標を表すと、ODT スムージングによる移動先は以下となる:

$$\frac{\sum_{j=1}^{n_i} |T_{i,j}| \mathbf{c}_{i,j}}{\sum_{j=1}^{n_i} |T_{i,j}|}. \quad (6)$$

DistMesh の移動法は頂点を均一に分布させるよう働く。3. でこの移動法を用い、頂点密度判定に式 4 の絶対的な辺長関数を採用したため、いずれ頂点密度が十分高まり、アルゴリズム 頂点追加は収束する。

2.3 結果

150 頂点の一様メッシュに対し提案アルゴリズムを施した結果、16 個の頂点が追加された (図 1(a))。非一様メッシュの例では、687 頂点から 749 頂点になった (図 1(b))。アルゴリズム適用後の各メッシュと、それぞれとほぼ同数の頂点を持つメッシュとの品質を比較した (表 1)。比較対象のメッシュは DistMesh で生成し、メッシュの種類に応じたスムージングをかけたものである。厳密に同じ頂点数でないのは DistMesh では頂点数の厳密な指定ができないためである。等方性は、各三角形の面積と辺長の二乗和の比で定義されるアスペクト比 (値域は $[0, 1]$ 、値が大きいほど等方的であることを示す) で測った。辺長関数との整合性を測る指標である辺長比は、辺 e_i に対して l_i^*/l_i ($l_i^* \leq l_i$) で、または l_i/l_i^* ($l_i < l_i^*$) で定義される。0 以上 1 以下の値をとり、1 に近いほど良い。アスペクト比の平

表 1. アルゴリズム 頂点追加を実行した結果と比較対象 (詳細は本文参照) のアスペクト比と辺長比の比較.

	一様メッシュ		非一様メッシュ	
	比較対象	提案手法	比較対象	提案手法
頂点数	169	166	751	749
アスペクト比 平均	0.970	0.975	0.977	0.981
アスペクト比 分散	1.60×10^{-3}	1.31×10^{-3}	7.81×10^{-4}	5.24×10^{-4}
辺長比 平均	—	—	0.938	0.945

均値が大きく、分散が小さくなっていることから、頂点追加でより等方的なメッシュが得られた事が分かる。非一様メッシュでは、辺長比の平均値も大きくなっている。これらの結果から提案アルゴリズムの有効性が確認できた。また、メッシュ内部の頂点の次数を調べた結果、2次元三角形メッシュで最適な次数6のものの割合が増加していた。

3 表面形状を再現する四面体メッシュ生成

本節では、曲面からなる境界をもつ3次元領域に対して、等方的な一様四面体メッシュを生成することを目的とする*1。第2節とは異なり、本節で提案するアルゴリズムは、等方性は多少低下しても境界との幾何誤差を減少する方針である。

3.1 四面体メッシュ生成

以下に提案するアルゴリズムを示す。頂点の移動や接続関係の変更は全て四面体メッシュに対してのみ行い、表面メッシュは一切変更しない。

アルゴリズム 四面体メッシュ生成

入力: 内部を四面体メッシュ生成領域とする表面メッシュ $\mathcal{M}_{\text{surf}}$, 四面体メッシュの頂点数を定めるパラメータ

出力: 等方的な四面体メッシュ \mathcal{M}

1. 初期四面体メッシュの生成
2. 頂点移動と接続関係の更新: 収束するまで以下を繰り返す。
 - 2-1. 領域内部に対する操作
 - 2-2. 領域境界に対する操作
3. 四面体メッシュ \mathcal{M} の抽出

初期四面体メッシュの生成: 入力で与えたパラメータに基づく細かさの格子で $\mathcal{M}_{\text{surf}}$ 内部を分割し、格子点を \mathcal{M} の初期頂点とする。初期頂点に対して Delaunay 単体分割を行う。

領域内部に対する操作: ODT スムージングで生成領域の内部の頂点を動かし、全頂点に対し Delaunay 単体分割を行うことを収束するまで繰り返す。ODT スムージングを用いるのは、境界を除けば ODT スムージングは接続する単体を2次元なら正三角形、3次元なら正四面体に近づけようとするので等方的なメッシュ生成に適するからである (但し正四面体では3次元空間を充填できない)。

領域境界に対する操作: まず領域境界を表現する頂点を決定する。次にそれらの頂点を後述の quadric error metric (QEM) を用いて決定した位置に移動し、Delaunay 単体分割を行う。

境界を表現する頂点は、以下のように選ぶ [1]: $\mathcal{M}_{\text{surf}}$ の各頂点が \mathcal{M} の頂点で最も近い位置にあるものを選ぶとき、 $\mathcal{M}_{\text{surf}}$ の少なくとも一つの頂点に選ばれた \mathcal{M} の頂点を、境界を表現する頂点とする。境界を表現する頂点 v_i (座標 (x_i, y_i, z_i)) を選んだ $\mathcal{M}_{\text{surf}}$ の頂点集合 $V_{\mathcal{M}_{\text{surf}}}(v_i)$ は、 v_i の Voronoi 領域と $\mathcal{M}_{\text{surf}}$ の共通部分の離散的な近似になっている。 v_i の、表面メッシュ $\mathcal{M}_{\text{surf}}$ に対する誤差 (QEM) を

$$\Delta'(v_i) = \tilde{v}_i^T \left(\sum_f |f| \mathbf{f} \cdot \mathbf{f}^T / \sum_f |f| \right) \tilde{v}_i \quad (7)$$

と定義する。ここで $\tilde{v}_i = (x_i, y_i, z_i, 1)^T$ であり、 \mathbf{f} は $\mathcal{M}_{\text{surf}}$ の面 f (面積 $|f|$) を含む平面 $ax + by + cz + d = 0$ から定

まるベクトル $(a, b, c, d)^T$ である。式7の和はどちらも、頂点 $v_j \in V_{\mathcal{M}_{\text{surf}}}(v_i)$ に接続する面集合に関してとる。アルゴリズム 四面体メッシュ生成 3-1. では、誤差 $\Delta'(v_i)$ が大きい順に次の手順で頂点 v_i を移動する。

アルゴリズム 領域境界を表現する頂点 v_i の位置決定

1. QEM を最小にする位置 v_i^* の算出
2. 頂点 v_i の移動: v_i^* を、 v_i の移動先候補 v_i^\dagger の初期値とする。 v_i^\dagger に v_i を移動するかどうかを決定する。以下の基準を共に満たす時のみ v_i^\dagger に v_i を移す。
 - v_i^\dagger に v_i を移動後、接続する要素の等方性の平均値と最小値の低下が、移動前のものに対して共に一定の割合未満である
 - v_i と v_i^\dagger との距離が一定以下である
基準を満たさなかった場合には、現在の位置 v_i と v_i^* を結ぶ線分上で二分探索を行う。この線分の中点を改めて移動先候補 v_i^\dagger とし判定を行うことを、 v_i^\dagger が v_i に十分近くなるか移動先が決定するまで繰り返す。等方性は内接球の半径と外接球の半径の比で測る。
3. 接続関係の更新: Delaunay 単体分割をする

メッシュ抽出: Delaunay 四面体分割は点集合の凸包に対する分割であるため、四面体集合の中からメッシュ生成領域を表現するものを選ぶ必要がある。 \mathcal{M} の要素として抽出する四面体は、重心が $\mathcal{M}_{\text{surf}}$ 内部にあり、かつ、外心から $\mathcal{M}_{\text{surf}}$ までの距離の外接球の半径に対する比が十分小さいものとする [1]。

3.2 結果

アルゴリズム 四面体メッシュ生成に対する計算機実験の結果が図2である。表面メッシュ $\mathcal{M}_{\text{surf}}$ (図2右半分に、 \mathcal{M} に重ねて表示) の頂点集合と \mathcal{M} の表面上の頂点集合とのハウスドルフ距離は 8.40×10^{-2} 、QEM の平均値は 2.17×10^{-6} であった。

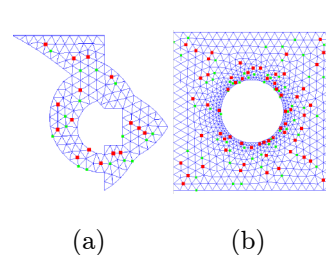


図 1. 頂点追加後のメッシュの例。大きい四角が新たに追加された頂点、小さい四角が周囲に頂点が少ないと判断された頂点を表す。(a) 一様メッシュ、(b) 非一様メッシュに提案手法を施した結果、それぞれ 16 頂点、62 頂点追加された。

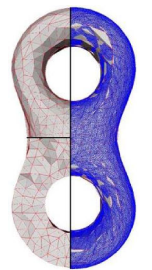


図 2. 左上: 四面体メッシュ \mathcal{M} . 左下: \mathcal{M} の断面. 右: $\mathcal{M}_{\text{surf}}$ のワイヤフレームと \mathcal{M} を重ねて表示したもの。

参考文献

- [1] Pierre Alliez, David Cohen-Steiner, Marianne Yvinec, and Mathieu Desbrun. Variational tetrahedral meshing. *ACM Transactions on Graphics*, Vol. 24, No. 3, pp. 617–625, 2005.
- [2] Long Chen. Mesh smoothing schemes based on optimal Delaunay triangulations. In *Proceedings of the 13th International Meshing Roundtable*, pp. 109–120, 2004.
- [3] Per-Olof Persson and Gilbert Strang. A simple mesh generation in MATLAB. *SIAM review*, Vol. 46, No. 2, pp. 329–345, 2004.

*1 本節のアルゴリズムは第2節とは全く独立なものである。