

# 頑健かつ高速な等角球体パラメータ化計算手法

金 井 崇<sup>†</sup>

メッシュの球体パラメータ化は、メッシュを球体に写像する技術であり、コンピュータグラフィックスや CAD/CAM などの分野での様々な応用が期待されている。本論文では、中でも品質の高い等角球体パラメータ化に関して、頑健かつ高速な計算手法を提案する。本手法では、メッシュの多重解像度表現に基づき、その階層構造をパラメータ化計算に利用する。従来手法に比べて高速であり、また球体パラメータを頑健かつ安定して求めることができることを様々な実験を通じて実証する。

## Robust and Fast Calculation of Conformal Spherical Parameterization

TAKASHI KANAI<sup>†</sup>

Spherical parameterization is one of geometry processing techniques which maps a mesh to a sphere, and is expected to the use in the various applications of computer graphics, CAD/CAM and so on. In this paper, we propose a robust and fast computation method for high quality conformal spherical parameterization. Our approach is based on using multi-resolution representation of a mesh. We utilize such a hierarchical mesh structure to the computation of parameterization. We demonstrate through several experiments that our approach is faster than the previous approaches and enables robust and stable computation of spherical parameterization.

### 1. はじめに

近年の設計・製造分野やコンピュータグラフィックスの分野での三次元立体形状モデリングにおいて、三角形メッシュ（以下、メッシュと呼ぶ）による形状処理の研究はさかに行われており、一部の技術はすでに実用化されている。このようなメッシュによるモデリングのことを総称してデジタル幾何処理（*digital geometry processing*）と呼んでいる。特に、三次元測定機から得られる大規模な幾何データに関するデジタル幾何処理は、基礎技術だけでなく、用途の広さから様々な応用面においてもその発展を期待されている。

パラメータ化（*parameterization*）は、メッシュからより単純なプリミティブ（たとえば平面や球など）へ写像するための技術であり、デジタル幾何処理の中の 1 つの基礎技術として認知されている。パラメータ化を用いたデジタル幾何処理の他の応用技術はすでに数多く行われており、たとえば、テクスチャマッピング、リメッシング、モーフィングなどがそれにあたる。これらのパラメータ化技術の中でも、球体パラメータ化（*spherical parameterization*）は、種数 0 のメッシュを球体に写像する技術であり、その処理の手軽さから近年特に注目されている。

本研究では、球体パラメータ化の中でも品質が良いとされている、等角球体パラメータ化（*conformal spherical parameterization*<sup>8)</sup>）について、頑健かつ高速に計算するための手法を提案する。パラメータ化における品質を決める 1 つの尺度として、等角性（*conformity*）の高い、すなわち、もとのメッシュの面の角度を保存する写像が良いとされている<sup>2),3),5)</sup>。図 1 に、パラメータ化における品質の違いを比較した結果を示す。図の右側が等角球体パラメータ化による結果であり、メッシュ上に歪みのないテクスチャが貼られていることが確認できる。また、ここでいう「頑健」ということは、写像の際メッシュ上の点と球体上の点が 1 対 1 対応になっていることを意味する。図 2 は、パラメータ化計算が失敗したときの一例であり、球体パラメータの一部の面が裏返っている（*flipping*）状態を示している。Gu らによって示された解法<sup>8)</sup>は、解の収束に必要な係数の設定が困難であり、計算時間がかかるという問題点を持つ。本研究では、上記のすべての問題を解決する、実用的な解法を提案するものである。

図 1 に、パラメータ化における品質の違いを比較した結果を示す。図の右側が等角球体パラメータ化による結果であり、メッシュ上に歪みのないテクスチャが貼られていることが確認できる。また、ここでいう「頑健」ということは、写像の際メッシュ上の点と球体上の点が 1 対 1 対応になっていることを意味する。図 2 は、パラメータ化計算が失敗したときの一例であり、球体パラメータの一部の面が裏返っている（*flipping*）状態を示している。Gu らによって示された解法<sup>8)</sup>は、解の収束に必要な係数の設定が困難であり、計算時間がかかるという問題点を持つ。本研究では、上記のすべての問題を解決する、実用的な解法を提案するものである。

<sup>†</sup> 慶應義塾大学環境情報学部

Faculty of Environmental Information, Keio University

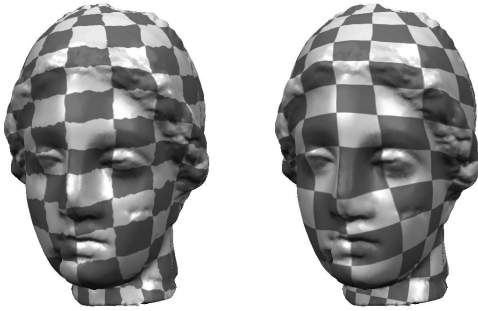


図 1 球体パラメータ化を利用したテクスチャマッピングの例．左：Alexa の手法<sup>1)</sup>による球体パラメータ，右：等角球体パラメータ

Fig. 1 Texture mapping examples using spherical parameterization. Left: spherical embedding by Alexa's method, Right: spherical embedding by conformal mapping.

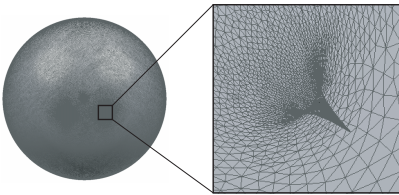


図 2 球体パラメータ化における面の裏返し

Fig. 2 Faces are flipped (*flipping*) on spherical parameterization.

本論文の構成は以下のとおりである．2 章には関連研究を示す．3 章には等角球体パラメータ化について述べ，4 章では本研究で提案するアルゴリズムを示す．5 章ではいくつかの例に対する実験結果について考察し，最後に 6 章で結論と展望を述べる．

## 2. 関連研究

メッシュのパラメータ化の最新動向に関しては Floater らによるサーベイ論文<sup>5)</sup> に詳しく述べられている．

平面へのパラメータ化に関しては過去に数多くの研究が行われており，特に文献 2)～4) では疎の線形連立一次方程式を解くことで得られるため，高速かつ頑健に計算することが可能である．ただし，円盤と位相同型のパッチ形状にのみ適用可能であることから，メッシュの形状全体をパラメータ化するには，アトラス (*atlas*) と呼ばれるパッチへの分割処理が必要となる<sup>13),18)</sup>．このことは，パッチ間の境界付近において，パラメータが不連続になるという問題を引き起こす原因となる（ごく最近，Khodakovsky らによってパッチの不連続の問題を解決する手法が提案されている<sup>12)</sup>）．

一方，種数 0 のメッシュを対象とした球体パラメータ化は，扱える対象形状が限定されているものの，上記の問題を解決する方法として注目されている．古くは，Kent らが 3D モーフィングの中で 2 つの形状間の対応をとるために球体へのパラメータ化を提案した<sup>11)</sup>．ただしこの方法は，球への直接的な投影に基づく方法であり，対象となる形状が星型形状に限定されていた．Haker らは，いったん立方体や八面体にパラメータ化し，それを球体に再パラメータ化する手法を提案した<sup>9)</sup> が，上記のパッチの不連続の問題が起きてしまううえに処理が煩雑である．Gotsman らは平面へのパラメータ化における重心写像を球体へ拡張した<sup>7)</sup>．しかしながら，この方法は非線形連立方程式を解くこととなり，安定した計算を行うには適切なソルバを選択する必要がある．さらに Praun らは，メッシュの伸縮度の最小化 (*stretch-minimizing*) に基づく球体パラメータ化について提案した<sup>14)</sup>．ここでは階層化手法を採用しており，本手法で用いる階層的な手法はこの手法にならっている．ただし，この手法は放物線最小化手法に基づく Brent の手法<sup>15)</sup> を利用することから，その最適化計算はいささか煩雑である．

Alexa は 3D モーフィングの中でよりシンプルな収束計算に基づく手法を提案した<sup>1)</sup>．Gu らは同様の手法で等角写像を実現した<sup>8)</sup>．本研究では Gu らの手法を用いることを基本とする．

いずれにせよ，これらの研究では，特に大容量のメッシュに対して，計算が不安定であったり，また収束に時間がかかったりすることから，とても実用的な方法とはいえないのが現状である．

## 3. 等角球体パラメータ化

本章では，等角球体パラメータ化と Gu らによるアルゴリズム<sup>8)</sup> について議論する．等角球体パラメータ化のもととなる等角写像 (*conformal mapping*)<sup>5),8)</sup> は次のように定義される：メッシュ  $M$  から球体  $S^2$  への写像  $f: M \mapsto S^2$  を定義する．この写像  $f$  が等角写像であるとき，それぞれの対応する点における第 1 基本行列 (*first fundamental matrix*) の各要素  $\mathbf{I}_{ij}^M, \mathbf{I}_{ij}^{S^2}$  ( $i = 1, 2, j = 1, 2$ ) に対し，以下の等式が成り立つ：

$$\mathbf{I}_{ij}^M = \mu(u, v) \mathbf{I}_{ij}^{S^2}, \quad (u, v) \in S^2, \quad \mu(u, v) \neq 0$$

ここで  $\mu$  はスカラー関数である．

Gu らによる等角球体パラメータ化アルゴリズム<sup>8)</sup> では，上記の式を離散化し，2 つのエネルギーを最小化する最適化問題として定式化している．

Tutte エネルギー  $E$  , 等角エネルギー  $F$  は以下のよう表される .

$$E = \sum_e \|\mathbf{x}(u) - \mathbf{x}(v)\|^2 \quad (1)$$

$$F = \sum_e (a_{v,u}^\alpha + a_{v,u}^\beta) \|\mathbf{y}(u) - \mathbf{y}(v)\|^2 \quad (2)$$

ここで  $u$  は  $v$  に隣接する頂点である . また ,  $a_{v,u}^\alpha$  ,  $a_{v,u}^\beta$  はエッジ  $e = \{v, u\}$  に隣接する 2 つの面  $\alpha$  ,  $\beta$  の角度に関する量である . たとえば面  $\alpha = \{v, u, w\}$  に対する  $a_{v,u}^\alpha$  は以下のように表せる .

$$a_{v,u}^\alpha = \frac{1}{2} \frac{(\mathbf{x}(v) - \mathbf{x}(w)) \times (\mathbf{x}(u) - \mathbf{x}(w))}{|(\mathbf{x}(v) - \mathbf{x}(w)) \times (\mathbf{x}(u) - \mathbf{x}(w))|} \quad (3)$$

ここで  $\mathbf{x}(u)$  ,  $\mathbf{x}(v)$  ,  $\mathbf{x}(w)$  は頂点  $u$  ,  $v$  ,  $w$  の三次元座標を示す .

このアルゴリズムは最急降下法に基づき , Tutte 球体パラメータ化と等角球体パラメータ化からなる以下の 2 つの手続きにより計算される .

#### Tutte 球体パラメータ化

- (1) ガウスマップ  $N: M \rightarrow S^2$  を計算し , これを各頂点  $v \in M$  における入力  $\mathbf{x}(v) = N(v)$  とする . Tutte エネルギー  $E_0$  を計算する .
- (2) 各頂点  $v$  に対し , 導関数  $D\mathbf{x}(v)$  を計算する .
- (3)  $\mathbf{x}(v)$  を  $\delta\mathbf{x}(v) = -D\mathbf{x}(v)\delta\mathbf{x}$  により更新し  $\mathbf{x}'(v)$  とする .
 
$$\mathbf{x}'(v) = \mathbf{x}(v) + c_t \delta\mathbf{x}(v) \quad (4)$$
 ここで  $c_t$  は Tutte 球体パラメータ化における収束定数である .
- (4) Tutte エネルギー  $E$  を計算する . もし  $|E - E_0|$  がある閾値 (ここでは  $1.0 \times 10^{-5}$ ) 内ならば処理を終了する . そうでなければ ,  $E_0 \leftarrow E$  ,  $\mathbf{x}(v) \leftarrow \mathbf{x}'(v)$  として (2) に戻る .

#### 等角球体パラメータ化

- (1) 上の球体パラメータ化による結果  $\mathbf{x}(v)$  を入力  $\mathbf{y}(v) = \mathbf{x}(v)$  とする . 等角エネルギー  $F_0$  を計算する .
- (2) 各頂点  $v$  に対し , 導関数  $D\mathbf{y}(v)$  を計算する .
- (3)  $\mathbf{y}(v)$  を  $\delta\mathbf{y}(v) = -D\mathbf{y}(v)\delta\mathbf{y}$  により更新し  $\mathbf{y}'(v)$  とする .
 
$$\mathbf{y}'(v) = \mathbf{y}(v) + c_c \delta\mathbf{y}(v) \quad (5)$$
 ここで  $c_c$  は等角球体パラメータ化における収束定数である .

- (4) メビウス変換 における拘束の代用として以下の重心保存処理を行う .
  - (a) 重心  $\mathbf{c} = \sum_v \mathbf{y}'(v)$  を計算する .
  - (b) 各頂点  $v$  に対し ,  $\mathbf{y}''(v) = \mathbf{y}'(v) - \mathbf{c}$  とする .
  - (c) 各頂点  $v$  に対し ,  $\mathbf{y}''(v)$  を正規化する .
- (5) 等角エネルギー  $F$  を計算する . もし  $|F - F_0|$  がある閾値 (ここでは  $1.0 \times 10^{-5}$ ) 内ならば処理を終了する . そうでなければ ,  $F_0 \leftarrow F$  ,  $\mathbf{y}(v) \leftarrow \mathbf{y}''(v)$  として (2) に戻る .

導関数  $D\mathbf{x}(v)$  ,  $D\mathbf{y}(v)$  は以下のように表される :

$$D\mathbf{x}(v) = \Delta\mathbf{x}(v) - (\Delta\mathbf{x}(v) \cdot \mathbf{x}(v)) \mathbf{x}(v) \quad (6)$$

$$D\mathbf{y}(v) = \Delta\mathbf{y}(v) - (\Delta\mathbf{y}(v) \cdot \mathbf{y}(v)) \mathbf{y}(v) \quad (7)$$

ここで ,  $(\mathbf{a} \cdot \mathbf{b})$  は  $\mathbf{a}$  と  $\mathbf{b}$  の内積を示す . ラプラシアン  $\Delta\mathbf{x}(v)$  および  $\Delta\mathbf{y}(v)$  は以下のように計算する .

$$\Delta\mathbf{x}(v) = \sum_{e=\{v,u\}} (\mathbf{x}(u) - \mathbf{x}(v)) \quad (8)$$

$$\Delta\mathbf{y}(v) = \sum_{e=\{v,u\}} (a_{v,u}^\alpha + a_{v,u}^\beta) (\mathbf{y}(u) - \mathbf{y}(v)) \quad (9)$$

以上のように , このアルゴリズムは収束計算となる . また , Tutte 球体パラメータ化は , 等角球体パラメータ化の初期値を算出するという役割を持つ . この計算により , 等角球体パラメータ化の際よく近似されたパラメータから出発することができ , よって収束回数を減らすことができる .

各手続きにおいては , 収束定数  $c_t$  ,  $c_c$  を決定する必要がある . これらの定数は , 計算の速度と安定性を決定するための重要な要素である .

本アルゴリズムはシンプルであるが , 特に大容量メッシュに関する適用は困難である . なぜならば , 上記の収束定数を適切に設定することがきわめて難しく , 試行錯誤を必要とするからである . 定数を小さくとると , 計算における安定性は増すが , それだけ収束に時間がかかる . 逆に定数を大きくとると , 計算が不安定になる (具体的には , エネルギーが最小値に向かうのではなく逆に発散する) という問題が発生する . 我々の実験では , 滑らかな形状に関しては比較的安定に計算できるものの , 表面の凹凸が激しい形状に対しては小さい定数値 ( $1.0 \times 10^{-3}$  もしくはそれ以下) をとる必要があることを確認している .

ガウスマップとは , 頂点の法線の集合である . すなわち (1) は , 各頂点に対して法線を計算し , その法線を初期値とする , ということである .

メビウス変換とは ,  $f(z) = (az + b)/(cz + d)$  ,  $ad - bc \neq 0$  を満たす変換のことで , 角度球体パラメータ化もこの変換の一種である . パラメータの自由度が大きく (6 自由度) , 一般には唯一の解を持たないため , 拘束を追加する必要がある .

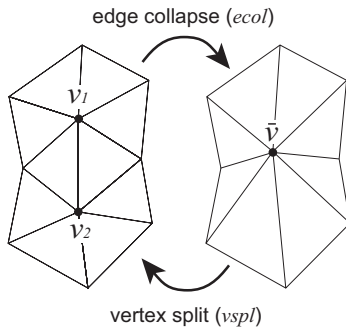


図 3 エッジ消去と頂点分割

Fig. 3 Edge collapse and vertex split.

収束定数を適切に決定するための 1 つの方法としては、まず、Tutte エネルギー  $E$  および等角エネルギー  $F$  の最小値をあらかじめ見積もっておく。収束計算の際、はじめは大きい値（たとえば  $1.0 \times 10^{-1}$  程度）で計算し、最小値の近くまできたときに小さい値（たとえば  $1.0 \times 10^{-3}$  程度）に再設定する。この方法では、最小値の見積りに数回程度の試行が必要となるため、形状が既知の場合には有効であるが、任意のメッシュに対する有効な手立てではない。また、他の方法として、収束定数を繰り返し回数に応じて段階的に小さくする、という方法も考えられる。しかしながら、形状によって収束回数はまちまちであるため、この方法も決定的な方法ではない。

#### 4. 頑健かつ高速な計算手法

本章では、前章までにあげた問題点をふまえ、「頑健」でかつ「高速」な球体パラメータ化の計算手法について提案する。本手法は、他の手法<sup>14),16),17)</sup>でも見られるようなメッシュの多重解像度表現を用いた階層的解法である。

##### 4.1 PM 表現による初期値の推定

階層的解法に利用するための基本ツールとして、本手法ではプログレッシブメッシュ表現（以下 PM 表現<sup>10)</sup>と呼ばれるメッシュの多重解像度表現を用いる。

Hoppe により提案された PM 表現<sup>10)</sup>では、エッジ消去（edge collapse）とその逆操作である頂点分割（vertex split）の 2 つの操作（図 3）により、任意の頂点数のメッシュを取り出すことができる。この PM 表現は、入力メッシュ  $M$  にエッジ消去を繰り返すメッシュ簡略化<sup>6)</sup>を適用することで変換することができる。

ここでは、PM 表現を主にパラメータ最適化の際の初期値算出のために用いる。はじめに粗いメッシュに対してパラメータを算出し、その後 PM 表現の各頂点分割操作ごとに、新しく生成された頂点のパラメータ

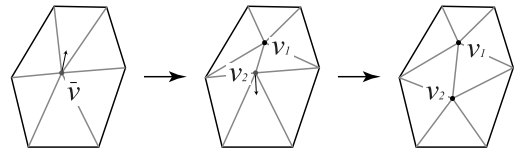


図 4 頂点分割操作におけるパラメータの計算手順

Fig. 4 Procedure for computing parameters in a vertex split operation.

を計算していく。その手順の様子を図 4 に示す。各頂点分割操作に対し、1 つの頂点  $\bar{v}$  が 2 つの頂点  $v_1$ ,  $v_2$  に分割される。このときまず  $v_1$  に関してパラメータ最適化を行い、次に  $v_2$  に対して行う。

個々のパラメータにおいて最適化計算を行うが、その詳細は次の節で説明する。

##### 4.2 頂点近傍におけるパラメータ最適化

本節では、頂点ごとのパラメータ最適化について説明する。この操作は、上記の初期値算出の際、より全体の最適解に近い初期値を計算するために用いる。また、後述する優先度付き待ち行列を用いたパラメータ全体の最適化の一部でも利用する。

頂点ごとの最適化計算として、頂点 Tutte 最適化、頂点等角最適化の 2 つを以下に定義する。

##### 頂点 Tutte 最適化

- (1) 頂点 Tutte エネルギー  $E_0(v)$  を計算する。
- (2) 頂点  $v$  に対し  $x'(v) = x(v) + c_t \delta x(v)$  としてパラメータを更新する。
- (3) 更新されたパラメータを用いて頂点 Tutte エネルギー  $E(v)$  を計算し、 $|E(v) - E_0(v)|$  がある閾値内になれば処理を終了する。
- (4)  $E_0(v) \leftarrow E(v)$ ,  $x(v) \leftarrow x'(v)$  として (2) に戻る。

##### 頂点等角最適化

- (1) 頂点 Tutte エネルギー  $F_0(v)$  を計算する。
- (2) 頂点  $v$  に対し  $y'(v) = y(v) + c_t \delta y(v)$  としてパラメータを更新する。
- (3) 更新されたパラメータを用いて頂点 Tutte エネルギー  $F(v)$  を計算し、 $|F(v) - F_0(v)|$  がある閾値内になれば処理を終了する。
- (4)  $F_0(v) \leftarrow F(v)$ ,  $y(v) \leftarrow y'(v)$  として (2) に戻る。

エネルギー  $E(v)$ ,  $F(v)$  はそれぞれ次式のとおりである。

$$E(v) = \sum_{u \in \nu} \|x(u) - x(v)\|^2 \quad (10)$$

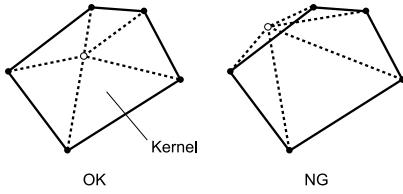


図 5 頂点  $v$  の核における内外判定

Fig. 5 Inside/outside classification in the kernel of a vertex  $v$ .

$$F(v) = \sum_{u \in \nu} (a_{v,u}^\alpha + a_{v,u}^\beta) \|y(u) - y(v)\|^2 \quad (11)$$

ここで  $\nu$  は  $v$  の近傍頂点の集合である。すなわち、 $E(v)$ 、 $F(v)$  は式 (1)、式 (2) のエネルギーを頂点近傍単位に置き換えたものにほかならない。

頑健なパラメータ計算を行うために、上記の最適化計算ごとにパラメータのチェックを行う。図 5 のように、ある頂点  $v$  の近傍の頂点群より構成される空間を頂点  $v$  の核 (kernel) と呼ぶことにする。新たに計算される頂点のパラメータがこの核の中に存在すれば、パラメータの埋め込みは維持され、その結果パラメータ計算は「頑健」であるといえる。本手法では、このパラメータのチェックを頂点のパラメータ最適化後に行い、もし核の外にはみ出すようであれば (図 5 右)、分割前の頂点のパラメータ値に戻している。

#### 4.3 優先度付き待ち行列を用いた全パラメータ最適化

球体パラメータ化アルゴリズムの過程の中で数回程度、パラメータ全体の最適化計算を行う。この全パラメータ最適化の計算のために、3 章で説明したアルゴリズムのかわりに、以下のような優先度付き待ち行列を利用したアルゴリズムを別途定義する。

##### 全パラメータ最適化

- (1) メッシュ  $M^n$  の各頂点に対し、頂点等角最適化前後のエネルギー  $F_0(v)$ 、 $F(v)$  を計算し、 $|F(v) - F_0(v)|$  ( $F(v) - F_0(v)$  の絶対値) をキーとして優先度付き待ち行列に挿入する。
- (2) 優先度付き待ち行列の先頭の (キーが最大となる) 要素を取り出し、最適化を実行する (頂点のパラメータを最適化後のものに置き換える)。
- (3) 先頭の要素のキーがある閾値内であれば (本手法では  $1.0 \times 10^{-8}$ ) 処理を終了する。
- (4) その要素の頂点の各隣接頂点に対し、頂点等角最適化を行いキーを更新する。(2) に戻る。

この方法の利点として、最適化計算によるエネルギーの変化の差の大きい頂点パラメータを集中して更新することから、全頂点のパラメータを 1 度に 1 回

ずつ更新する前節の方法に比べて効率が良いと思われる。これより、収束計算の繰返し回数の減少につながることが期待される。

#### 4.4 階層的アルゴリズム

提案する階層的な球体パラメータ化アルゴリズムは以下のとおりである。

##### 階層的球体パラメータ化アルゴリズム

- (1) 粗いメッシュ  $M^m$  に対し、球体パラメータ化を行い、このパラメータを出発点とする。本手法では、この計算に Alexa の手法<sup>1)</sup>を用いている。また最初の頂点数  $m$  は 100 から 500 の間でとっている。
- (2) PM 表現の頂点分割操作を行い、新しく生成される 2 つの頂点に対して頂点 Tutte 最適化を施す。この操作を、メッシュの頂点数がもとの  $n$  倍になるまで続けて行う。本手法では  $n = 2$  としている。
- (3) メッシュ  $M^{n \times m}$  (頂点数  $n \times m$  のメッシュ) に対し全パラメータ最適化を施す。 $M^m \leftarrow M^{n \times m}$  として (2) に戻る。
- (4) (2) の過程の中で、もとのメッシュ  $M$  になった場合、そこで全パラメータ最適化を 1 度施して終了する。

粗いメッシュに対するパラメータ化の際、以下の 2 点の理由により Alexa の手法を用いている。1. 我々の実験より、頂点数が少ない状態ではこの方法が最も安定して計算できる。2. この段階ではパラメータの初期値を計算するのが目的であり、パラメータの質はそれほど問題にならない。

また、(2) の頂点分割操作の際の最適化計算には頂点 Tutte 最適化のみを行っている。これは、等角球体パラメータ化の初期値には Tutte 球体パラメータ化の結果を用いる、というオリジナルの手法に従っている。実際、(2) が終了した段階でのパラメータの初期値は、Tutte 球体パラメータ化の結果に非常に近いものになっている。

#### 5. 実験結果と考察

本章では、様々な条件のもとに球体パラメータ化手法を適用し、その結果について検証・考察を行う。なお、本実験はすべて Pentium 4 3.2 GHz CPU の計算機上で GNU g++によりコンパイルしたコードを用いて時間を計測した。

まず初めに、1 つ目の実験として、Gu らの手法と本手法との比較のため、図 6 左の triceratops のメッ

表 1 triceratops メッシュにおける様々な収束定数のもとでの Gu らの手法<sup>8)</sup> の統計結果  
 Table 1 Statistics with various iteration coefficients in Gu, et al.'s approach<sup>8)</sup> for triceratops mesh.

|     | coeffs.              |                      | embed. | time (sec.) |       |        | num. iter. |        | conf. energy $F$ |
|-----|----------------------|----------------------|--------|-------------|-------|--------|------------|--------|------------------|
|     | $c_t$                | $c_c$                |        | tutte       | conf. | total  | tutte      | conf.  |                  |
| (1) | $5.0 \times 10^{-2}$ | $5.0 \times 10^{-3}$ | ×      | -           | -     | -      | -          | -      | -                |
| (2) | $3.0 \times 10^{-2}$ | $3.0 \times 10^{-4}$ | ×      | 19.1        | -     | -      | 6,302      | -      | -                |
| (3) | $3.0 \times 10^{-2}$ | $2.0 \times 10^{-4}$ |        | 19.0        | 129.9 | 148.9  | 6,302      | 11,146 | 25.99            |
| (4) | $2.0 \times 10^{-2}$ | $2.0 \times 10^{-4}$ |        | 23.1        | 121.3 | 144.4  | 7,669      | 10,559 | 26.08            |
| (5) | $1.0 \times 10^{-2}$ | $1.0 \times 10^{-4}$ |        | 40.8        | 155.1 | 195.9  | 13,443     | 13,490 | 26.48            |
| (6) | $5.0 \times 10^{-3}$ | $1.0 \times 10^{-4}$ | ×      | 73.6        | 161.1 | 234.7  | 24,332     | 14,004 | 26.59            |
| (7) | $1.0 \times 10^{-3}$ | $1.0 \times 10^{-4}$ | ×      | 235.4       | 171.7 | 407.1  | 77,385     | 15,054 | 28.04            |
| (8) | $1.0 \times 10^{-4}$ | $1.0 \times 10^{-4}$ | ×      | 1034.6      | 314.0 | 1348.6 | 320,058    | 27,431 | 40.47            |
| (9) | $1.0 \times 10^{-4}$ | $1.0 \times 10^{-5}$ | ×      | 961.1       | 403.1 | 1364.2 | 320,058    | 35,607 | 45.35            |

表 2 triceratops メッシュにおける階層的解法に基づく等角球体パラメータ化計算の統計結果  
 Table 2 Statistics of hierarchical conformal spherical parameterizations for triceratops mesh.

|     | coeffs               |                      | hier. | init. | optim. | embed. | total time (sec.) | conf. energy $F$ |
|-----|----------------------|----------------------|-------|-------|--------|--------|-------------------|------------------|
|     | $c_t$                | $c_c$                |       |       |        |        |                   |                  |
| (a) | $2.0 \times 10^{-2}$ | $2.0 \times 10^{-4}$ | no    | prog. | orig.  |        | 372.1             | 27.50            |
| (b) | $2.0 \times 10^{-2}$ | $2.0 \times 10^{-4}$ | yes   | prog. | orig.  |        | 177.2             | 25.60            |
| (c) | $2.0 \times 10^{-2}$ | $2.0 \times 10^{-4}$ | no    | prog. | pri.   |        | 176.0             | 25.95            |
| (d) | 1.0                  | 1.0                  | yes   | prog. | pri.   |        | 57.3              | 25.59            |

表 3 Armadillo メッシュにおける Alexa の手法 (Alexa)<sup>1)</sup>, Gu らの手法 (Gu, et al.<sup>8)</sup>, 本手法 (Hier. Conf.) の計算結果

Table 3 Statistics of Alexa's approach (Alexa)<sup>1)</sup>, Gu, et al.'s approach (Gu, et al.<sup>8)</sup> and our approach (Hier. Conf.) for Armadillo mesh.

| type            | $c_t$                | $c_c$                | hier. | init. | optim. | embed. | time (sec.) | conf. energy |
|-----------------|----------------------|----------------------|-------|-------|--------|--------|-------------|--------------|
| (1) Alexa       | -                    | -                    | no    | orig. | orig.  | ×      | 1,389.0     | -            |
| (2) Gu et al.   | $2.0 \times 10^{-1}$ | $1.0 \times 10^{-3}$ | no    | orig. | orig.  |        | 5,518.7     | 25.96        |
| (3) Gu et al.   | $1.0 \times 10^{-1}$ | $1.0 \times 10^{-3}$ | no    | orig. | orig.  | ×      | 8,385.3     | 28.60        |
| (4) Hier. Conf. | $1.0 \times 10^{-1}$ | $1.0 \times 10^{-3}$ | no    | prog. | orig.  |        | 20,370.5    | 28.00        |
| (5) Hier. Conf. | $1.0 \times 10^{-1}$ | $1.0 \times 10^{-3}$ | yes   | prog. | orig.  |        | 3,861.1     | 25.53        |
| (6) Hier. Conf. | $1.0 \times 10^{-1}$ | $1.0 \times 10^{-3}$ | no    | prog. | pq.    |        | 2,065.0     | 26.08        |
| (7) Hier. Conf. | 1.0                  | 1.0                  | yes   | prog. | pq.    |        | 830.1       | 25.17        |

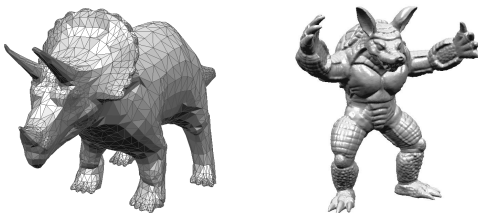


図 6 実験に使用したモデル。左: triceratops (頂点数 2,832), 右: Armadillo (頂点数 172,974)

Fig. 6 Meshes used in our experiments. Left: triceratops (vertices: 2,832), Right: Armadillo (vertices: 172,974).

シュ (頂点数: 2,832) に対し, Gu らの手法を様々な収束定数の下で実行した。その結果を表 1 に示す。この表は, 収束係数  $c_t$ ,  $c_c$  を変化させた計 9 パターンの計算結果の違いを表している。表より, 計算に成功

した (embed. で となっている) パターンは計 3 つしかなく, 残りの 6 つのパターンは計算に失敗している。このうち, 最初の 2 つのパターン ((1)-(2)) は, エネルギーが発散してしまい, 正常に終了しなかった。また, 残りの 4 つのパターン ((6)-(9)) は, エネルギーは収束するものの, 図 2 のように面が裏返ってしまう結果となった。これより,  $c_t$  が  $3.0 \times 10^{-2} \sim 1.0 \times 10^{-2}$ ,  $c_c$  が  $2.0 \times 10^{-4} \sim 1.0 \times 10^{-4}$  と, 非常に狭い数値の範囲でしか計算に成功しないことが見てとれる。このことは, Gu らの手法の計算の難しさを示している: すなわち, 計算に成功するための収束係数の調整を試行錯誤によって行う必要がある。

係数を小さくすればするほど, 収束にかかる回数 (num. iter.) が次第に増えているのが分かる。これは, 一度に計算される差分がそれだけ小さくなってしまふことが原因である。また, 等角エネルギー (conf.

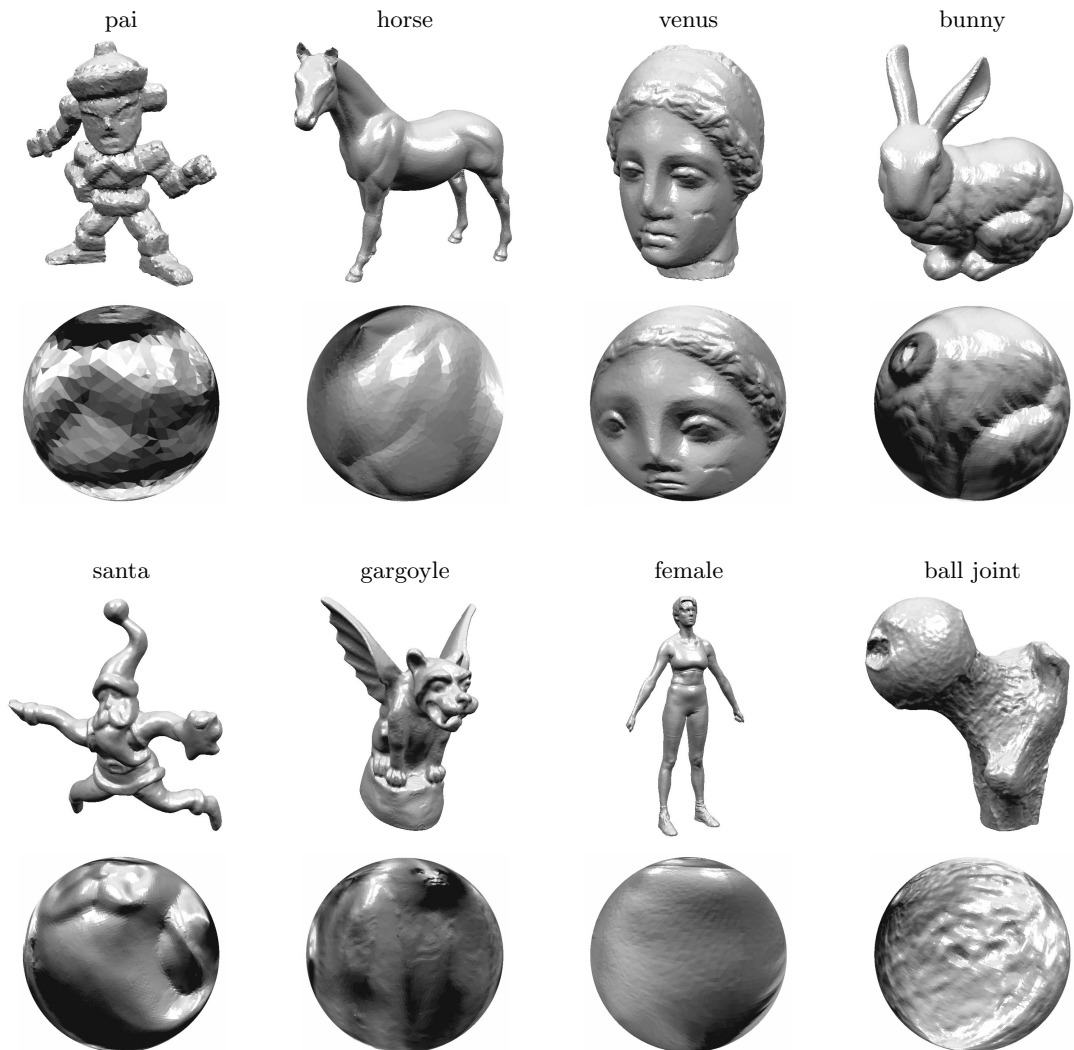


図 7 様々なメッシュにおける本手法の適用結果．上段：メッシュ，下段：球体パラメータ  
 Fig. 7 Results of our approach to various meshes, Top: A mesh. Bottom: Its spherical embedding.

energy, 式 (2) の  $F$ ) は収束が正しく行われているかどうかを示す尺度である．等角エネルギーが小さければより最小値に近くなる．

一方、同じ triceratops メッシュに対し、本手法で提案する階層的解法を適用した場合の結果を表 2 に示す．この表では計 4 つのパターンによる計算結果の違いについて示している．これら 4 つのパターンはいずれも計算に成功している．表の hier. は、階層的な全パラメータ化 (4.4 節のアルゴリズムの (3)) の有無、init. は初期値推定の種類 (prog. は PM 表現による初期値推定 (4.4 節のアルゴリズムの (1)), orig. は Gu らにおける初期値推定) を、optim. は、最適化手法の違い (orig. は Gu らの手法、pri. は 4.3 節の全

パラメータ最適化) を示している．

表の (a) と (b) より、階層的な全パラメータ化を施すことにより、計算時間が減少し、かつエネルギーが小さくなっているのが分かる．これは、初期推定値が最適値により近似されているので、各段階における繰返し回数がより少なくて済むからである．また表の (a), (c) より、優先度付き待ち行列を用いた全パラメータ最適化による効果として、繰返しの回数が減少したことにより、その分高速化につながっている．これらをすべて組み合わせた (d) は、表 1 の最も短い計算時間に比べておおよそ 1/3 弱、エネルギーは最も小さい値となっている．また収束係数  $c_t, c_c$  についても、表 1 の例よりも大きい値をとることができ、その分計算時



表4 図7のメッシュに対する計算時間の比較．なお，すべての例において  $c_t = 1.0$  ,  $c_c = 1.0$  とした  
 Table 4 Comparisons of computation time for meshes in Fig. 7. We set  $c_t = 1.0$  ,  
 $c_c = 1.0$  for all examples.

|                   | pai    | horse  | venus  | bunny   | santa  | gargoyle | female  | ball joint |
|-------------------|--------|--------|--------|---------|--------|----------|---------|------------|
| #vertices         | 12,344 | 19,851 | 33,591 | 34,817  | 75,781 | 100,002  | 121,723 | 137,062    |
| total time (sec.) | 100.0  | 206.3  | 228.7  | 1,322.3 | 354.3  | 907.2    | 719.4   | 923.3      |

間の減少に貢献している．これより，本手法の高速化に関する寄与が確実に表れていることが確認できる．なお， $c_t$  ,  $c_c$  の上限値は，式(4)，式(5)において差分を本来の計算値よりも大きくとることができないことから1.0である．実際，1.0より大きい値をとると計算が発散することが実験により分かった．

2番目の実験として，図6右の Armadillo メッシュ(頂点数: 172,974)に対し，Alexaの手法，Guらの手法，および本手法を適用し考察を行った．その結果を表3に示す．Armadilloメッシュは頂点数が多く，そのため本手法の特徴がより如実に表れる結果となっている．Guらの手法は，実験の一例を載せたが，実験1と同様収束係数の調整にかなりの時間を要した．Alexaの手法は，Guらの手法よりも高速であるが，それでも，本例では計算には成功しなかった．本手法における計算時間は，収束係数を調整したGuらの手法に比べても6.5倍程度高速であり，本手法の特徴である高速性が，特に頂点数の多いメッシュに対して有効であることが確認できる．さらに，本手法は計算にすべて成功しており，頑健な計算手法であることも確認できる．

最後に，本手法を様々なメッシュに対して適用した結果を図7に，その計算時間の比較を表4に示す．頂点数はまちまちであるが，すべてのメッシュに対して頑健に計算できることを確認した．また計算時間について，一概に頂点数が多いメッシュが計算時間が多くなるわけではないことが分かる．これは，メッシュの幾何形状によって初期値の近似の度合いが異なり，その結果，繰返し回数が増減することが一因である．

## 6. おわりに

本論文では，等角球体パラメータ化について，階層的解法に基づく手法を提案した．様々な実験を通じて，本手法が高速かつ頑健な計算手法であり，より実用的な手法であることを実証した．

今後の課題としては，1つに制約を課した場合の計算手法についての確立があげられる．これは，特にモーフィングにおいて球体パラメータ化を用いるときには必須の技術である．もう1つの課題として，この球体パラメータ化を用いたさらなるアプリケーション

についても考えていきたい．

謝辞 本実験に使用したデータのうち Armadillo , bunny はスタンフォード大学，horse , venus , santa , female , ball joint は Cyberware 社，pai は東京大学，triceratops は Viewpoint 社，gargoyle は Microsoft 社のデータである．また，本研究の一部は(財)大川情報通信基金によるご支援をいただいた．ここに感謝の意を表する．

## 参考文献

- 1) Alexa, M.: Morphing Polyhedral Shapes with Scattered Features, *The Visual Computer*, Vol.16, No.1, pp.26-37 (2000).
- 2) Desbrun, M., Meyer, M. and Alliez, P.: Intrinsic Parameterizations of Surface Meshes, *Computer Graphics Forum (Eurographics 2002 Proceedings)*, Vol.21, No.3, pp.209-218 (2002).
- 3) Eck, M., DeRose, T., Duchamp, T., Hoppe, H., Lounsbery, M. and Stuetzle, W.: Multiresolution Analysis of Arbitrary Meshes, *Computer Graphics (Proc. SIGGRAPH 95)*, pp.173-182, ACM Press, New York (1995).
- 4) Floater, M.S.: Parametrization and Smooth Approximation of Surface Triangulations, *Computer Aided Geometric Design*, Vol.14, No.4, pp.231-250 (1997).
- 5) Floater, M.S. and Hormann, K.: Recent Advances in Surface Parameterization, *Proc. Multiresolution in Geometric Modelling 2003*, pp.259-284, Springer-Verlag, Berlin (2003).
- 6) Garland, M. and Heckbert, P. S.: Surface Simplification Using Quadric Error Metrics, *Computer Graphics (Proc. SIGGRAPH 97)*, pp.209-216, ACM Press, New York (1997).
- 7) Gotsman, C., Gu, X. and Sheffer, A.: Fundamentals of Spherical Parameterization for 3D Meshes, *ACM Trans. Graphics (Proc. SIGGRAPH 2003)*, Vol.22, No.3, pp.358-363 (2003).
- 8) Gu, X., Wang, Y., Chan, T.F., Thompson, P.M. and Yau, S.-T.: Genus Zero Surface Conformal Mapping and Its Application to Brain Surface Mapping, *IEEE Trans. Medical Imaging*, Vol.23, No.8, pp.949-958 (2004).
- 9) Haker, S., Angenent, S., Tannenbaum, A.,



- Kikinis, R., Sapiro, G. and Halle, M.: Conformal Surface Parameterization for Texture Mapping, *IEEE Trans. Visualization and Computer Graphics*, Vol.6, No.2, pp.181–189 (2000).
- 10) Hoppe, H.: Progressive Meshes, *Computer Graphics (Proc. SIGGRAPH 96)*, pp.99–108, ACM Press, New York (1996).
- 11) Kent, J.R., Carlson, W.E. and Parent, R.E.: Shape Transformation for Polyhedral Objects, *Computer Graphics (Proc. SIGGRAPH 92)*, pp.47–54, ACM Press, New York (1992).
- 12) Khodakovsky, A., Litke, N. and Schröder, P.: Globally Smooth Parameterizations with Low Distortion, *ACM Trans. Graphics (Proc. SIGGRAPH 2003)*, Vol.22, No.3, pp.350–357 (2003).
- 13) Lévy, B., Petitjean, S., Ray, N. and Maillot, J.: Least Squares Conformal Maps for Automatic Texture Atlas Generation, *ACM Trans. Graphics (Proc. SIGGRAPH 2002)*, Vol.21, No.3, pp.362–371 (2002).
- 14) Praun, E. and Hoppe, H.: Spherical Parametrization and Remeshing, *ACM Trans. Graphics (Proc. SIGGRAPH 2003)*, Vol.22, No.3, pp.340–349 (2003).
- 15) Press, W.H., Teukolsky, S.A., Vetterling, W.T. and Flannery, B.P.: *Numerical recipes in C*, 2nd edition, chapter 10.2 Parabolic Interpolation and Brent's Method in One Dimension, pp.402–405, Cambridge University Press, Cambridge, UK (1992).
- 16) Ray, N. and Lévy, B.: Hierarchical Least Squares Conformal Map, *Proc. Pacific Graphics 2003*, pp.263–270, IEEE CS Press, Los Alamitos, CA (2003).
- 17) Sander, P.V., Gortler, S.J., Snyder, J. and Hoppe, H.: Signal-Specialized Parametrization, *Proc. 13th Eurographics Workshop on Rendering*, pp.87–98, Eurographics Association, Aire-la-Ville, Switzerland (2002).
- 18) Sander, P.V., Snyder, J., Gortler, S.J. and Hoppe, H.: Texture Mapping Progressive Meshes, *Computer Graphics (Proc. SIGGRAPH 2001)*, pp.409–416, ACM Press, New York (2001).

(平成 16 年 5 月 20 日受付)

(平成 16 年 12 月 1 日採録)



金井 崇 (正会員)

昭和 44 年生 . 平成 10 年東京大学大学院工学系研究科精密機械工学専攻博士課程修了 . 同年理化学研究所基礎科学特別研究員 . 平成 12 年慶應義塾大学環境情報学部専任講師 .

現在に至る . 平成 14 年理化学研究所ものづくり情報技術統合化研究プログラム非常勤研究員兼任 . 形状モデリングの CAD/CAM および CG への応用に関する研究に従事 . 博士 (工学) . 精密工学会 , 画像電子学会 , 芸術科学会 , ACM , IEEE CS 各会員 . 平成 13 年度山下記念研究賞受賞 .