

多重解像度制約を用いた細分割曲面設計における局所平滑化

1 E-2-3

高橋 成雄

群馬大学総合情報処理センター

1 はじめに

従来、複雑な曲面形状の設計は、適当なパラメータ曲面表現を用いて行い、さらにそこから離散的にサンプル点をとって多面体曲面 (polygonal surface) 表現に変換し、ディスプレイ上に表示していた。ところが、パソコンに代表されるように、近年の計算機性能の向上は、多面体曲面を直接操作する環境をもたらしようになった。また多面体曲面は、パラメータ曲面表現では連続なパッチ接続の問題で難しい任意位相の曲面を、柔軟に表現することができる。多面体曲面は、レーザ・スキャナなどの計測装置から得られる密なサンプル点の表現にも都合がよく、形状表現の標準的な道具として広く研究がなされてきている。さらに、膨大なデータ量を持つ多面体曲面を適宜簡単化して、階層表現を導入する手法も提案されている [1]。

多面体曲面の中で、細分割曲面 (subdivision surface) [2, 3, 4, 5] は形状設計の分野においていち早く注目を集めていて、幾何形状モデリングに広く応用されてきていた。また、多面体曲面の場合と同様に、位相的な細分割規則によって生じる細分割曲面の系列を、曲面の階層表現ととらえる手法の提案もなされている。Lounsbery ら [6] は、信号処理の道具であるウェーブレット (wavelet) を導入し、細分割曲面の多重解像度表現を構築することに成功した。また、細分割曲面の多重解像度表現を利用してその形状を設計する手法も提案されており、中でも Zorin ら [7] は、Taubin [8] の局所平滑化フィルタを用いて細分割曲面を階層的に表現し、さらに部分部分適応的に細分割を制御できる設計手法を提案した。

我々は、先の論文 [9] で Lounsbery [6] らの細分割曲面の階層表現を用いて、その曲面形状を設計する手法を提案している。Zorin らの手法を含め、従来手法は複数の

解像度レベルの形状を同時に制御できなかったのに比べ、我々の手法は、異なる解像度レベルに異なる幾何制約を付加して、同時に形状を制御することができる点が優れている。さらに我々の手法では、解像度レベルごとに重みを割り当て、その重みの値を制御することで、形状の滑らかさを変更することができることも特筆すべき点である。しかし、この手法には Welch と Witkin の手法 [10, 11] に基づく、細分割曲面の頂点数の2乗の線形方程式を解くグローバルな形状最適化が含まれていて、非常に計算コストが高いことが問題であった。そこで本論文では、Taubin の局所平滑化フィルタ [8] を用い、このグローバルな形状最適化で実現された、解像度レベルごとの滑らかさの重みづけを効率よく模倣できる手法を構築する。Taubin の局所平滑化フィルタは、細分割曲面の頂点数に比例するオーダーで計算ができるため、従来手法に比べ提案手法は飛躍的に計算時間を短縮する。

2 細分割曲面上の多重解像度解析

まず、Lounsbery ら [6] によって提案された、ウェーブレットを用いた細分割曲面の多重解像度表現について説明する。多重解像度解析においては、形状を表す関数を $V^{(0)} \subset V^{(1)} \subset \dots \subset V^{(K)}$ となる、順々に入れ子になる関数空間の列 $V^{(k)}$ ($k = 0, \dots, K$) を構成することができる。ここで、添字 (k) は解像度レベル k を表している。 $V^{(k)}$ の基底関数を、解像度レベル k のスケール関数と呼び、その基底関数を並べたベクトルを $\varphi^{(k)} = (\varphi_0^{(k)} \dots \varphi_{m^{(k)}}^{(k)})^T$ と表すことにする。ただし、 $m^{(k)}$ は、 $V^{(k)}$ の次元である。さらに、 $W^{(k)}$ を $V^{(k+1)}$ と $V^{(k)}$ の差となるように、つまり直和 $\oplus$ を用いて $V^{(k+1)} = V^{(k)} \oplus W^{(k)}$ と表されるようにとる。このとき、 $W^{(k)}$ の基底関数をウェーブレットと呼び、その基底関数を並べたベクトルを $\psi^{(k)} = (\psi_0^{(k)} \dots \psi_{n^{(k)}}^{(k)})^T$ と表すことにする。ただし、 $n^{(k)}$ は、 $W^{(k)}$ の次元である。

本論文では、Lounsbery らによって定式化された、二重直交ウェーブレットを用いる。半直交ウェーブレットの条件の方が、 $V^{(k)}$ と $W^{(k)}$ が直交するという点で望ま

しいが、これを細分割曲面上に課すと、ウェーブレットの台が曲面全体に広がってしまう。そのため、半直交の条件を二重直交の条件に緩めて、ウェーブレットの局所的な台を実現するのである。

関数 $C^{(k)} \in V^{(k)}$ と $D^{(k)} \in W^{(k)}$ を考えると、それらは $C^{(k)} = \varphi^{(k)\top} \cdot c^{(k)}$ と $D^{(k)} = \psi^{(k)\top} \cdot d^{(k)}$ と、それぞれスケール関数とウェーブレットの線形和として書くことができる。ここで、 $c^{(k)}$ と $d^{(k)}$ は、対応する係数のベクトルを表していることに注意する。 $V^{(k)} \subset V^{(k+1)}$ かつ $W^{(k)} \subset V^{(k+1)}$ であるから、次を満たす行列 $P^{(k)}, Q^{(k)}$ が存在する。

$$\varphi^{(k)}(x)^\top = \varphi^{(k+1)}(x)^\top P^{(k)} \quad (1)$$

$$\psi^{(k)}(x)^\top = \varphi^{(k+1)}(x)^\top Q^{(k)} \quad (2)$$

行列 $P^{(k)}, Q^{(k)}$ は、合成フィルタと呼ばれる。ウェーブレットに基づく階層表現は、係数ベクトル $c^{(k)}, d^{(k)}$ に線形変換を施すことで得ることができる。まず、ウェーブレット分解は、

$$c^{(k)} = A^{(k)} c^{(k+1)}, \quad d^{(k)} = B^{(k)} c^{(k+1)} \quad (3)$$

と表現される。ここで、(3) の $A^{(k)}, B^{(k)}$ は、分解フィルタと呼ばれる。逆にウェーブレット再構成は、

$$c^{(k+1)} = P^{(k)} c^{(k)} + Q^{(k)} d^{(k)} \quad (4)$$

となり、(1),(2) と同じ行列 $P^{(k)}, Q^{(k)}$ で表現できる。また、分解・合成フィルタの間の、次の関係を導くことができる。

$$\begin{pmatrix} A^{(k)} \\ B^{(k)} \end{pmatrix} = \begin{pmatrix} P^{(k)} & Q^{(k)} \end{pmatrix}^{-1} \quad (5)$$

3 細分割規則

細分割曲面は、初期制御多面体 $\Theta^{(0)}$ を、 $\Theta^{(1)}, \Theta^{(2)}, \dots$ へと再帰的に細分割していったときの、最終的に収束する曲面 $\Theta^{(\infty)}$ として定義される。 $c^{(k)}$ を、 $\Theta^{(k)}$ の頂点のひとつの座標 (例えば x, y, z 座標のいずれか) を要素とするベクトルとすると、細分割規則から

$$c^{(k+1)} = P^{(k)} c^{(k)} \quad (6)$$

の関係が得られる。ここで、 $m^{(k)}$ は $\Theta^{(k)}$ の頂点数を表し、 $P^{(k)}$ は $m^{(k+1)} \times m^{(k)}$ 行列となる。ゆえに、スケール関数は曲面上のいずれかの頂点に 1 対 1 に対応し、その個数も頂点の数だけ存在することになる。のちの分解・合成フィルタの計算に備えて、スケール関数のベクトルを、 $\varphi^{(k+1)\top} = (\mathcal{O}^{(k+1)\top} \mathcal{N}^{(k+1)\top})$ のように分割

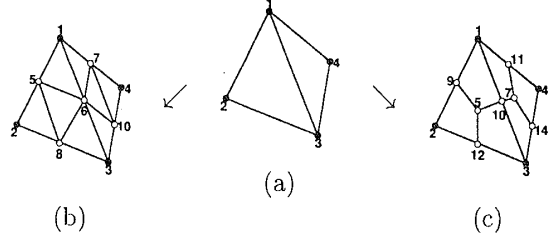


図 1: 三角形分割と四角形分割規則: (a) 四面体 (b) 三角形分割規則を施したもの (c) 四角形分割規則を施したもの。

しておく。ここで、ベクトル $\mathcal{O}^{(k+1)}(x)$ は、スケール関数 $\varphi_\alpha^{(k+1)}(x)$ ($\alpha = 1, \dots, m^{(k+1)}$) のうち、 $\Theta^{(k)}$ に元々存在した頂点に対応するものを要素にとり、 $\mathcal{N}^{(k+1)}(x)$ は、 $(k+1)$ 回目の細分割処理で新しく生じた頂点に対応するスケール関数を要素とする。これより、(1) を

$$\varphi^{(k)\top} = \begin{pmatrix} \mathcal{O}^{(k+1)\top} & \mathcal{N}^{(k+1)\top} \end{pmatrix} \begin{pmatrix} \mathcal{O}^{(k)} \\ \mathcal{N}^{(k)} \end{pmatrix} \quad (7)$$

のように書き直すことができる。ここで、 $\mathcal{O}^{(k)}$ と $\mathcal{N}^{(k)}$ は $P^{(k)}$ の一部分であり、それぞれ $\mathcal{O}^{(k+1)}(x)$ と $\mathcal{N}^{(k+1)}(x)$ に対応している。

本論文では、三角形細分割と四角形細分割規則を用いる。図 1 は、四面体が、三角形細分割と四角形細分割規則を用いて、どのように分割されるかを示したものである。これらは、もとの多面体の面を形状を変えずに、平面的に分割するものであり、それぞれ位相的には Loop[4] と Catmull-Clark[2] 細分割曲面に等しい。本手法では、細分割を行うごとに、新しく生じた頂点の位置を局所平滑化を用いて決めるため、このような平面的な細分割規則を用いても、最終的に滑らかな曲面を生成することができる。

4 分解・合成フィルタの計算

Lounsbery ら [6] は、分解・合成フィルタを、スケール関数やウェーブレットの形を明示的に求めることなく、細分割規則から得られる線形連立方程式を解くことで求めている。そのためには、細分割曲面上のスケール関数同士の内積を要素にもつ行列 $I^{(k)}$ を計算する必要がある。まず内積を、細分割曲面上の面の形や大きさに依存しないように、以下のように定義する。

$$\langle F^{(k)}, G^{(k)} \rangle \equiv \sum_{\tau \in \Delta(\Theta^{(0)})} \frac{1}{\text{Area}(\tau)} \int_{x \in \tau} F^{(k)}(x) G^{(k)}(x) dx \quad (8)$$

ここで、 $\Delta(\Theta^{(0)})$ は、制御多面体 $\Theta^{(0)}$ 上の多面体の集合を表して、 $\text{Area}(\tau)$ は、多面体 τ の面積を表している。

先に述べた通り、 $I^{(k)}$ は (i, j) 成分が $\langle \varphi_i^{(k)}, \varphi_j^{(k)} \rangle$ であるような、 $m^{(k)} \times m^{(k)}$ 行列であるとする。細分割規則から、次のような $I^{(k)}$ と $I^{(k+1)}$ の間の関係が導かれる。

$$I^{(k)} = P^{(k)\top} I^{(k+1)} P^{(k)} \quad (9)$$

細分割は局所的に行われるので、(9) は 0 でないスケール関数の内積に関する、連立線形方程式となり、結果として $I^{(k)}$ を計算することができる。

$I^{(k)}$ が得られると、二重直交ウェーブレットのための分解・合成フィルタは、以下のように計算できる。まず、 $m^{(k)} \times n^{(k)}$ 行列 $\alpha^{(k)} = (\alpha_{i,j}^{(k)})$ を考える。また、 $\varphi_\alpha^{(k+1)}$ ($\alpha = 1, \dots, n^{(k)}$) を、 $\Theta^{(k+1)}$ に新しく生成された頂点に対応するスケール関数であるとする。ここで、ウェーブレット $\psi_\alpha^{(k)}$ ($\alpha = 1, \dots, n^{(k)}$) が $\varphi_\alpha^{(k)}$ の最小二乗射影であるとするとき、すべての $\gamma \in \rho_\alpha^{(k)}$ に関して、以下が満たされる。

$$\sum_{\beta \in \rho_\alpha^{(k)}} \alpha_{\beta,\alpha}^{(k)} \langle \varphi_\beta^{(k)}(x), \varphi_\gamma^{(k)}(x) \rangle = \langle \varphi_\alpha^{(k+1)}(x), \varphi_\gamma^{(k)}(x) \rangle \quad (10)$$

この式(10)を用いて、 $\alpha^{(k)}$ を計算する。ここで、 $\langle \varphi_\beta^{(k)}(x), \varphi_\gamma^{(k)}(x) \rangle, \langle \varphi_\alpha^{(k+1)}(x), \varphi_\gamma^{(k)}(x) \rangle$ は、それぞれ $I^{(k)}, I^{(k+1)}$ より、求めることができることに注意する。また、 $\rho_\alpha^{(k)}$ は、 $\psi_\alpha^{(k)}$ の局所台を表して、あらかじめ決めておくことができる [6]。

分解・合成フィルタは、 $N^{(k)}$ と $\alpha^{(k)}$ から求めることができる。特に、本手法においては $O^{(k)} = 1$ であるので、それぞれのフィルタは、以下のように簡単に書くことができる [6]。

$$\begin{pmatrix} P^{(k)} & Q^{(k)} \end{pmatrix} = \begin{pmatrix} 1 & -\alpha^{(k)} \\ N^{(k)} & 1 - N^{(k)} \alpha^{(k)} \end{pmatrix} \quad (11)$$

$$\begin{pmatrix} A^{(k)} \\ B^{(k)} \end{pmatrix} = \begin{pmatrix} 1 - \alpha^{(k)} N^{(k)} & \alpha^{(k)} \\ -N^{(k)} & 1 \end{pmatrix} \quad (12)$$

5 幾何制約による形状制御

この節では、ある解像度レベル k の細分割曲面 $\Theta^{(k)}$ に与えられた制約を、どのように定式化するかを説明する。本研究で用いる幾何制約は、Welch と Witkin [10] による定式化であり、それらは有限次元制約と超限次元制約に分類される。有限次元制約は、点や微分などのある離散パラメータ値における制約である。例えば、曲面 $\Theta^{(k)}$ をあるパラメータ値 x_0 において点 θ_0 に固定する制

約は、

$$\Theta^{(k)}(x_0) = \varphi^{(k)}(x_0)^\top \cdot c^{(k)} = \theta_0 \quad (13)$$

と書くことができる。超限次元制約は、曲線や曲面などの連続パラメータ領域における制約である。例えば、パラメータ曲線 $l(t)$ に課せられた曲線制約 $L(l(t))$ は、次のように書くことができる。

$$\left(\int \varphi^{(k)}(l) \cdot \varphi^{(k)}(l)^\top dt \right) c^{(k)} = \int L \cdot \varphi^{(k)}(l) dt. \quad (14)$$

これより、どちらの制約も最終的には $c^{(k)}$ に関する、線形連立方程式になる。

$$M^{(k)} c^{(k)} = r^{(k)}. \quad (15)$$

細分割曲面上にどのようにパラメータを導入するかも、考える必要がある。ここでは、細分割曲面上のそれぞれの多角形に重心座標を導入する。例えば、点 v_0 が三角形 $\Delta v_i^{(k)} v_j^{(k)} v_k^{(k)}$ に含まれていて、 $v_0 = t_i v_i^{(k)} + t_j v_j^{(k)} + t_k v_k^{(k)}$ の式が成り立つとする。このとき、その点のパラメータ値は、

$$(0 \cdots 0 t_i 0 \cdots 0 t_j 0 \cdots 0 t_k 0 \cdots 0), \quad (16)$$

と書くことにする。ただし、式(16)において、 t_i, t_j, t_k の値は i, j, k 番目の要素として現れることに注意する。一般的な多角形の重心座標については、[12] を参照されたい。

6 細分割曲面の平滑化

6.1 局所平滑化フィルタリング

前に述べたように、我々の前の手法 [9] は、制約を満たす滑らかな形状を求めるのに、多面体曲面の頂点数の 2 乗の線形方程式を解く必要があった。しかしこの手法は、パラメータ曲面の制御点よりもかなり多くの頂点を含む多面体曲面を扱う際には、非常にコストの高い計算となる。このコストの高い計算を避けるために、本論文では Taubin の局所平滑化フィルタ [8] を用いる。この手法は、頂点数に比例する計算量で、滑らかな多面体曲面形状を求めることができる点が優れている。

Taubin の手法は、形状の収縮を回避しながら Gaussian フィルタを low-pass フィルタに拡張することで実現される。まず、 i 番目の頂点の (x, y, z) いずれかの座標を、Gaussian フィルタで平滑化する。

$$v_i' = v_i + p \Delta v_i \quad (17)$$

ここで、 p は、 $0 < p < 1$ を満たすようなスケール因子である。また、 Δv_i は隣接する頂点との差を平均した値 $\Delta v_i = \frac{1}{n} \sum_{j=0}^n (v_j - v_i)$ であり、 $j (= 1, \dots, n)$ は隣接する頂点の番号を表している。Gaussian フィルタだけでは形状の収縮が生じるので、次のような負の Gaussian フィルタを用いて、形状を膨らませる。

$$v_i'' = v_i' + q\Delta v_i' \quad (18)$$

ここで、 q は負のスケール因子で、 $q < -\lambda$ を満たす。Taubin の論文 [8] では、スケール因子 p と q は、 $\frac{1}{p} + \frac{1}{q} = 0.1$ を満たすように選ばれている。

6.2 線形制約の扱い

Taubin は、彼の論文 [8] で頂点の座標に課せられた線形連立方程式を扱う一手法を示している。彼の手法は、ひとつの解像度レベルの線形制約に対してはよく機能するが、その制約が多重解像度レベルに渡るものだと、あまり良い結果を示さない。そこで、我々は次のような手法で、線形制約を満たす滑らかな形状を計算することを考える。ここでも、ある特定の解像度レベル k に限定して考え、さらに上付き添字 (k) を省略していく。

まず c_j が、 j 回平滑化処理を施した後の、細分割曲面の頂点のベクトルを表すとする。 $j+1$ 回目の平滑化処理により、 c_j が $\bar{c}_j$ に変更され、与えられた制約 $M\mathbf{c} = \mathbf{r}$ から少し逸脱することになる。ベクトル $\bar{c}_j$ が $M\mathbf{c} = \mathbf{r}$ を満たすようにするため、 $c_{j+1} = \bar{c}_j + \bar{c}_j$ を考え、方程式 $M\mathbf{c}_{j+1} = \mathbf{r}$ を最小二乗の意味で満たすように、 $\bar{c}_j$ を求める。 $M\mathbf{c}_{j+1} = \mathbf{r}$ であるから、 $\bar{c}_j$ は、

$$\bar{c}_j = (M^T M)^{-1} M^T (\mathbf{r} - M\bar{c}_j). \quad (19)$$

を解くことによって得られる。この処理を、十分滑らかな曲面が得られるまで続けるわけである。ここで、行列 $M^T M$ さらに $(M^T M)^{-1} M^T$ は、上の局所平滑化処理を繰り返す前に計算できることに注意する。行列 M は粗行列であるから、この行列計算は、線形方程式の数にほぼ依存する。

7 局所平滑化を用いた多重解像度制約の解法

この節では、多重解像度の曲面に与えられた制約を、局所平滑化を用いてどのように解くかを説明する。ここで注意すべき点は、本提案手法が、我々が前の論文 [9] で実現した、解像度レベル毎の滑らかさの重み付けの効果を、非常に効率よく模倣できる点であり、この点が本論文の主な貢献となる。

本手法の道すじとして、低解像度レベル 0 から高解像度レベル K まで各解像度レベルごとに、細分割曲面に形状の起伏を加えていくことによって、最終的な形状を決定していく。言い換えると、細分割を行うごとに曲面が指定した滑らかさになるように、その細分割処理で新しく生成された頂点の位置を決めていくのである。そこでまず、解像度レベル k で与えられる形状の起伏をベクトル $\mathbf{c}^{(k)}$ ($k = 0, \dots, K$) と書くことにする。この時、解像度レベル k の細分割曲面の頂点座標を表す $\mathbf{c}^{(k)}$ とは、図 2 のような関係があることに注意する [13]。

最初に考えなければならないのは、それぞれのレベル k に重み $\kappa^{(k)}$ ($k = 0, \dots, K$) が与えられたとき、 $\mathbf{c}^{(k)}$ ($k = 0, \dots, K$) にどのように重み付けを行えばよいかである。本手法では、付録 A の考察に基づき、他のレベルを考慮に入れないで求めた形状の

$$\frac{\kappa^{(k)}}{\kappa^\Sigma} \quad (20)$$

を $\mathbf{c}^{(k)}$ に割り当てることにする。ただし、

$$\kappa^\Sigma = \sum_{\lambda=0}^K \kappa^{(\lambda)} \quad (21)$$

とする。以下、各解像度レベル k ($k = 0, \dots, K$) において、線形制約方程式

$$M^{(k)} \mathbf{c}^{(k)} = \mathbf{r}^{(k)} \quad (22)$$

が課せられているとして、解像度レベルごとに、どのように形状の起伏を決定していくかを述べる。

まず、解像度レベル 0 を考える。解像度レベル 0 においては、そのレベルに課せられている制約方程式

$$M^{(0)} \mathbf{c}_f^{(0)} = \mathbf{r}^{(0)} \quad (23)$$

を満たすようなベクトル $\mathbf{c}_f^{(0)}$ を考える。実際に求める形状の起伏 $\mathbf{c}^{(0)}$ は、上で述べた重み付けの割り当てより、

$$\mathbf{c}^{(0)} = \frac{\kappa^{(0)}}{\kappa^\Sigma} \mathbf{c}_f^{(0)} \quad (24)$$

として求める。ここで、式 (23) と (24) より、 $\mathbf{c}^{(0)}$ に関する制約方程式

$$M^{(0)} \mathbf{c}^{(0)} = \frac{\kappa^{(0)}}{\kappa^\Sigma} \mathbf{r}^{(0)} \quad (25)$$

が得られるので、実際には (25) が満たされるように、 $\mathbf{c}^{(0)}$ に局所平滑化を施せばよい。その際、 $\mathbf{c}^{(0)}$ の初期値も重要になるが、今は解像度レベル 0 なので、

$$\mathbf{c}^{(0)} = \mathbf{0} \quad (26)$$

$$\begin{array}{ccccccc}
c^{(K)} & = & P^{(K-1)} \dots P^{(0)} c^{(0)} & + & P^{(K-1)} \dots P^{(1)} c^{(1)} & + & \dots + c^{(K)} \\
& & \uparrow & & \uparrow & & \downarrow \\
& & \vdots & & \vdots & & \vdots \\
& & \uparrow & & \uparrow & & \downarrow \\
c^{(1)} & = & P^{(0)} c^{(0)} & + & c^{(1)} & + & \dots + A^{(1)} \dots A^{(K-1)} c^{(K)} \\
& & \uparrow & & \downarrow & & \downarrow \\
c^{(0)} & = & c^{(0)} & + & A^{(0)} c^{(1)} & + & \dots + A^{(0)} \dots A^{(K-1)} c^{(K)}
\end{array}$$

図 2: ベクトル $c^{(k)}$ ($k=0, \dots, K$) と $c^{(k)}$ ($k=0, \dots, K$) の関係.

と初期値を設定する.

次に, 解像度レベル1について考える. 最初に, このレベル1で, 課せられた制約を満たす形状を考える. さらに, 求めた形状のうち解像度レベル0と1に相当する重みの割合を求め, さらにそこから既に得ている $c^{(0)}$ に対応する部分を取り除いて, $c^{(1)}$ の値とする. こうすることにより, 解像度レベル1に見合った細かさの起伏を, $c^{(1)}$ に含ませることができる.

具体的には, 最初に次の制約方程式

$$\begin{pmatrix} M^{(0)} A^{(0)} \\ M^{(1)} \end{pmatrix} c_f^{(1)} = \begin{pmatrix} r^{(0)} \\ r^{(1)} \end{pmatrix} \quad (27)$$

を満たすベクトル $c_f^{(1)}$ を考える. この方程式は, レベル0と1のそれぞれの制約方程式を [14] で提案した手法でまとめたものである. 解像度レベル1だけでなくレベル0の方程式を考慮に入れるのは, 解像度レベル0の制約を満たしたまま, 解像度レベル1の制約を満たす滑らかな形状を求めるためである. $c^{(1)}$ と $c_f^{(1)}$ の関係であるが, 先に述べた通り, $c_f^{(1)}$ に解像度レベル0と1に相当する重みの割合 $\frac{\kappa^{(0)} + \kappa^{(1)}}{\kappa^\Sigma}$ をかけたものから, $c^{(0)}$ に相当する部分を取り去ったものを $c^{(1)}$ とするので,

$$c^{(1)} = \frac{\kappa^{(0)} + \kappa^{(1)}}{\kappa^\Sigma} c_f^{(1)} - P^{(0)} c^{(0)} \quad (28)$$

となる. よって, 式 (27) と (28) より, $c^{(1)}$ が満たすべき方程式は,

$$\begin{pmatrix} M^{(0)} A^{(0)} \\ M^{(1)} \end{pmatrix} c^{(1)} = \frac{\kappa^{(0)} + \kappa^{(1)}}{\kappa^\Sigma} \begin{pmatrix} r^{(0)} \\ r^{(1)} \end{pmatrix} - \begin{pmatrix} M^{(0)} A^{(0)} \\ M^{(1)} \end{pmatrix} P^{(0)} c^{(0)} \quad (29)$$

となる.

解像度レベル0のときとは異なり, $c^{(1)}$ の初期値は慎重に選ばないと, 解像度レベル1に見合う細かさの起伏を $c^{(1)}$ に含めることができない. 初期値としては, 解像

度レベル1の起伏は含まれないが, レベル0の起伏は含まれているものが望ましい. このとき, 解像度レベル0までで計算した起伏は, 解像度レベル1では $P^{(0)} c^{(0)}$ と書くことができるので, 重みの割合を考慮すると,

$$\frac{\kappa^{(1)}}{\kappa^{(0)}} P^{(0)} c^{(0)} \quad (30)$$

を $c^{(1)}$ の初期値にすればよいことが分かる.

ここまで考えると, 一般的な解像度レベル k について, まとめることができる. まずベクトル $c_f^{(k)}$ が, 次のような方程式を満たすと考える.

$$\widehat{M}^{(k)} c_f^{(k)} = \widehat{r}^{(k)} \quad (31)$$

ただし,

$$\widehat{M}^{(k)} = \begin{pmatrix} M^{(0)} A^{(0)} \dots A^{(k-1)} \\ \vdots \\ M^{(k)} \end{pmatrix} \quad (32)$$

$$\widehat{r}^{(k)} = \begin{pmatrix} r^{(0)} \\ \vdots \\ r^{(k)} \end{pmatrix} \quad (33)$$

である. この方程式は, レベル k までの制約方程式をまとめたものである [14]. 次に, $c^{(k)}$ と $c_f^{(k)}$ の関係は, 重みの割合を考慮して,

$$c^{(k)} = \frac{\kappa^{(0)} + \dots + \kappa^{(k)}}{\kappa^\Sigma} c_f^{(k)} - \sum_{\lambda=0}^{k-1} P^{(k-1)} \dots P^{(\lambda)} c^{(\lambda)} \quad (34)$$

となればよい. 式 (34) の意味するところは, 方程式 (31) を満たすベクトル $c_f^{(k)}$ から, そのレベルまでの割合

$$\frac{\kappa^{(0)} + \dots + \kappa^{(k)}}{\kappa^\Sigma} \quad (35)$$

をかけて, そこから今まで求めた $c^{(0)}, \dots, c^{(k-1)}$ の分だけ取り除いたものを, $c^{(k)}$ として残すというものであ

る。ここで、式(31)と(34)より、

$$\widehat{M}^{(k)} c^{(k)} = \frac{\kappa^{(0)} + \dots + \kappa^{(k)}}{\kappa^\Sigma} \widehat{r}^{(k)} - \widehat{M}^{(k)} \left(\sum_{\lambda=0}^{k-1} P^{(k-1)} \dots P^{(\lambda)} c^{(\lambda)} \right) \quad (36)$$

のような $c^{(k)}$ に関する方程式が得ることができる。実際の平滑化は、(36) が満たされるように、 $c^{(k)}$ に対して行えばよい。この際、 $c^{(k)}$ には、解像度レベル k に見合った細かさの起伏が含まれることに注意する。

平滑化を行う際の $c^{(k)}$ の初期値についてもまとめる。解像度レベル 1 の初期値の選び方と同じように、今まで求めた解像度レベル $k-1$ までの起伏 $c^{(0)}, \dots, c^{(k-1)}$ の和に、適当な割合をかけたものを初期値として採用する。いまままで求めた起伏は、レベル k においては

$$\sum_{\lambda=0}^{k-1} P^{(k-1)} \dots P^{(\lambda)} c^{(\lambda)} \quad (37)$$

と書くことができる。それぞれの解像度レベル k には、 $\kappa^{(k)}$ の重みが割り当てられているから、 $c^{(k)}$ の実際の初期値には、

$$\frac{\kappa^{(k)}}{\kappa^{(0)} + \dots + \kappa^{(k-1)}} \sum_{\lambda=0}^{k-1} P^{(k-1)} \dots P^{(\lambda)} c^{(\lambda)} \quad (38)$$

を与えればよい。この初期値には、解像度レベル $k-1$ の起伏は含まれているが、レベル k の起伏は含まれていないことに注意する。

このようにして、解像度レベル 0 から K まで順々に局所平滑化を施して、 $c^{(k)}$ ($k = 0, \dots, K$) を求めればよい。

8 結果

図 3 と図 4 は、解像度レベル k の重みの値 $\kappa^{(k)}$ ($k = 0, \dots, K$) が異なると、形状の滑らかさがどのように変化するかを示している。ここで、このふたつの図では、

$$\kappa^{(k)} = \gamma^{k-K} \quad (39)$$

のように指数関数的に重みを割り当てていることに注意する。図 5 は、解像度レベル 4 の重み $\kappa^{(4)}$ だけを変更して、髪形の滑らかさを制御している例である。

参考文献

- [1] Hoppe, H., DeRose, T., Duchamp, T., McDonald, J. and Stuetzle, W.: Mesh Optimization, *Computer Graphics (Proceedings Siggraph '93)*, pp. 19-26 (1993).
- [2] Catmull, E. and Clark, J.: Recursively Generated B-spline Surfaces on Arbitrary Topological Meshes, *Computer-Aided Design*, Vol. 10, No. 6, pp. 350-355 (1978).
- [3] Doo, D. and Sabin, M.: Behaviour of Recursive Division Surfaces near Extraordinary Points, *Computer-Aided Design*, Vol. 10, No. 6, pp. 356-360 (1978).
- [4] Loop, C.: Smooth Subdivision Surfaces Based on Triangles, Master's thesis, University of Utah, Department of Mathematics (1987).
- [5] Dyn, N., Levin, D. and Gregory, J. A.: A Butterfly Subdivision Scheme for Surface Interpolation with Tension Control, *ACM Transactions on Graphics*, Vol. 9, No. 2, pp. 160-169 (1990).
- [6] Lounsbery, M., DeRose, T. D. and Warren, J.: Multiresolution Analysis for Surfaces of Arbitrary Topological Type, *ACM Transactions on Graphics*, Vol. 16, No. 1, pp. 34-73 (1997).
- [7] Zorin, D., Schröder, P. and Sweldens, W.: Interactive Multiresolution Mesh Editing, *Computer Graphics (Proc. Siggraph '97)*, pp. 259-268 (1997).
- [8] Taubin, G.: A Signal Processing Approach To Fair Surface Design, *Computer Graphics (Proc. Siggraph '95)*, pp. 351-358 (1995).
- [9] 高橋成雄: 細分割曲面のための多重解像度制約, *Visual Computing/グラフィクスとCAD 合同シンポジウム'99*, pp. 55-60 (1999).
- [10] Welch, W. and Witkin, A.: Variational Surface Modeling, *Computer Graphics (Proc. Siggraph '92)*, pp. 157-166 (1992).
- [11] Welch, W. and Witkin, A.: Free-Form Shape Design Using Triangulated Surfaces, *Computer Graphics (Proc. Siggraph '94)*, pp. 247-256 (1994).
- [12] Loop, C. T. and DeRose, T. D.: A Multisided Generalization of Bézier Surfaces, *ACM Transactions on Graphics*, Vol. 8, No. 3, pp. 204-234 (1990).
- [13] Takahashi, S.: Variational Design of Curves and Surfaces using Multiresolution Constraints, *The Visual Computer*, Vol. 14, No. 5/6, pp. 208-227 (1998).
- [14] Takahashi, S., Shinagawa, Y. and Kunii, T. L.: Curve and Surface Design using Multiresolution Constraints, *Computer Graphics International '97* (Vince, J. and Reeth, F. V. (eds.)), IEEE Computer Society Press, pp. 121-130 (1997).
- [15] Terzopoulos, D.: Regularization of Inverse Visual Problems Involving Discontinuities, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 8, No. 4, pp. 413-424 (1986).

A 薄板モデルからの近似

ここでは, Terzopoulos[15]の薄板モデルに基づき, 多重解像度制約[13]を解く際に, 求められる各解像度レベルの起伏 $c^{(k)}$ ($k = 0, \dots, K$) が, 解像度に割り当てられた重みの値とどのような関係になるかを, 考察する. 多重解像度制約のための薄板モデルの詳細については, [9]を参照のこと.

まず, $c^{(k)}$ をまとめて,

$$c_\sigma^\top = \begin{pmatrix} c^{(0)\top} & c^{(1)\top} & \dots & c^{(K)\top} \end{pmatrix} \quad (40)$$

と表す. このとき, $c^{(k)}$ と $c^{(k)}$ との関係は, 図2に示される通りである. 各解像度レベル k において, (22)の線形制約方程式が与えられているとすると, 図2の関係をを用いて次のような c_σ に関する方程式を得ることができる[9].

$$M_\sigma c_\sigma = r_\sigma \quad (41)$$

次に, 最小化すべき関数を考える. 各解像度レベル k ($k = 0, \dots, K$) において, $c^{(k)}$ に関するエネルギー関数は $c^{(k)\top} S^{(k)} c^{(k)}$ となる. ここで $S^{(k)}$ は, 上記関数が解像度レベル k の形状の起伏の曲げ・伸びを表すようにとられた正定値行列である[10]. 解像度レベル k のエネルギー関数に, $\frac{1}{\kappa^{(k)}}$ の重みが割り振られたとき, c_σ に関するエネルギー関数は, 満たすべき制約にラグランジュ乗数をかけたものと合わせて,

$$\frac{1}{2} c_\sigma^\top S_\sigma c_\sigma + (M_\sigma c_\sigma - r_\sigma)^\top y \quad (42)$$

となる. ただし,

$$S_\sigma = \begin{pmatrix} \frac{1}{\kappa^{(0)}} S^{(0)} & & 0 \\ & \ddots & \\ 0 & & \frac{1}{\kappa^{(K)}} S^{(K)} \end{pmatrix} \quad (43)$$

である. 結局 c_σ は, 以下の線形連立方程式を解いて求めることができる.

$$c_\sigma = K_\sigma J_\sigma^{-1} r_\sigma \quad (44)$$

ただし,

$$J_\sigma = M_\sigma S_\sigma^{-1} M_\sigma^\top, \quad K_\sigma = S_\sigma^{-1} M_\sigma^\top \quad (45)$$

である. また, 行列 S_σ の逆行列 S_σ^{-1} は,

$$S_\sigma^{-1} = \begin{pmatrix} \kappa^{(0)} S^{(0)-1} & & 0 \\ & \ddots & \\ 0 & & \kappa^{(K)} S^{(K)-1} \end{pmatrix} \quad (46)$$

と書くことができる.

ここで, 線形制約方程式がひとつしかない場合を考える. このとき, 制約を表す行列 M_σ とベクトル r_σ は, ベクトル m_σ とスカラー r になる. また, ベクトル m_σ の要素を, 各解像度レベルの形状を表す $c^{(k)}$ の次元にあわせて,

$$m_\sigma = (m_\sigma^{(0)} \dots m_\sigma^{(K)}) \quad (47)$$

と分割しておく. すると, 行列 $J_\sigma = M_\sigma S_\sigma^{-1} M_\sigma^\top$ は 1×1 行列, つまりスカラーになり

$$J_\sigma^{-1} = \frac{1}{m_\sigma S_\sigma^{-1} m_\sigma^\top} \quad (48)$$

となる. ここで,

$$m_\sigma S_\sigma^{-1} m_\sigma^\top = \sum_{k=0}^K \kappa^{(k)} m_\sigma^{(k)\top} S^{(k)-1} m_\sigma^{(k)} \quad (49)$$

となることに注意する. 式(44)より,

$$c^{(k)} = \frac{\kappa^{(k)} r S^{(k)-1} m_\sigma^{(k)}}{\sum_{k=0}^K \kappa^{(k)} m_\sigma^{(k)\top} S^{(k)-1} m_\sigma^{(k)}} \quad (50)$$

となる. ここで,

$$h^{(k)} = m_\sigma^{(k)\top} S^{(k)-1} m_\sigma^{(k)} (> 0) \quad (51)$$

とおくと,

$$c^{(k)} = \frac{\kappa^{(k)}}{\sum_{k=0}^K h^{(k)} \kappa^{(k)}} r S^{(k)-1} m_\sigma^{(k)\top} \quad (52)$$

さらに, $h^{(k)}$ がすべて同じ大きさになるという仮定をおいて簡単化を行うと,

$$c^{(k)} = \frac{\kappa^{(k)}}{\kappa^\Sigma} r S^{(k)-1} m_\sigma^{(k)\top} \quad (53)$$

となる. (κ^Σ の定義は, 式(21)を参照.) これより, 各解像度レベルで $\kappa^{(k)}$ ($k = 0, \dots, K$) の重みが割り当てられているとすると, そのレベル k の形状は, 他のレベルを考慮に入れないで求めた値の

$$\frac{\kappa^{(k)}}{\kappa^\Sigma} \quad (54)$$

の割合で近似することにする. 複数の制約が与えられたときには, 実際に解像度間の重みの割り当ての影響が複雑になるが, (54)の近似を基にして局所平滑化で求めた結果は, 薄板モデル下で得た結果を非常によく模倣する.

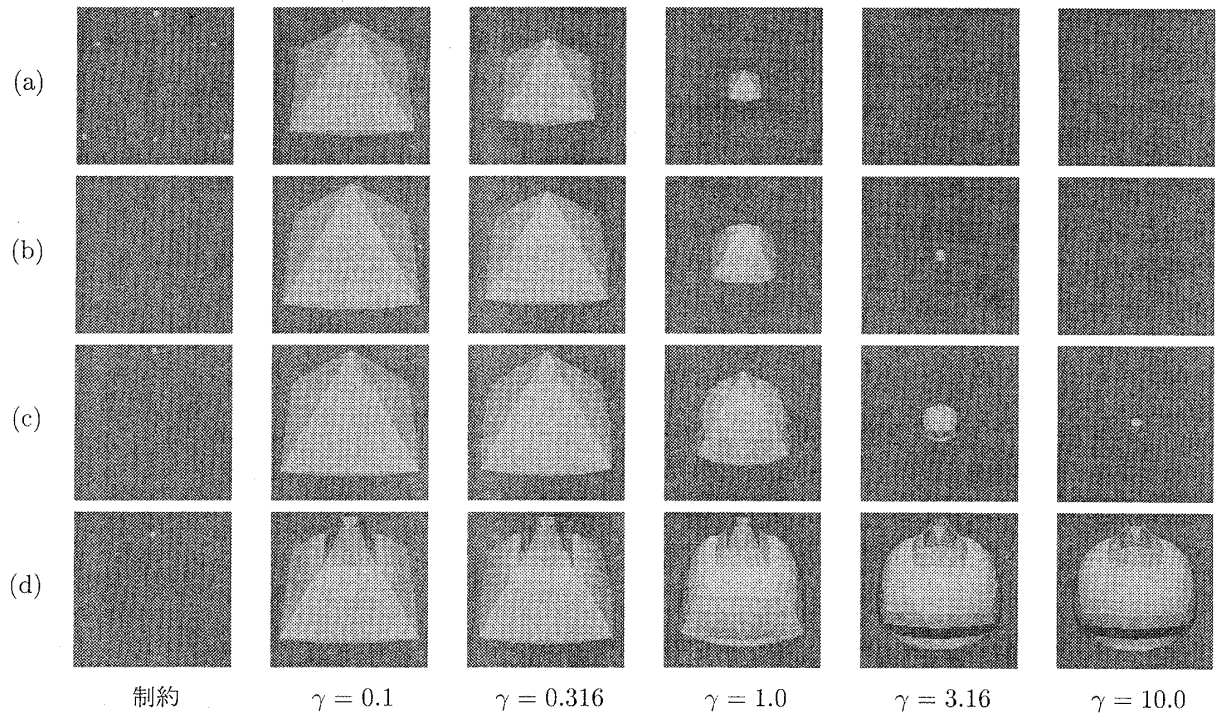


図 3: くぼみを設計する際の各解像度レベルの起伏の計算結果: (a) $c^{(0)}$, (b) $P^{(0)}c^{(0)} + c^{(1)}$, (c) $P^{(1)}(P^{(0)}c^{(0)} + c^{(1)}) + c^{(2)}$, (d) $P^{(2)}(P^{(1)}(P^{(0)}c^{(0)} + c^{(1)}) + c^{(2)}) + c^{(3)}$. γ の値は (39) を参照.

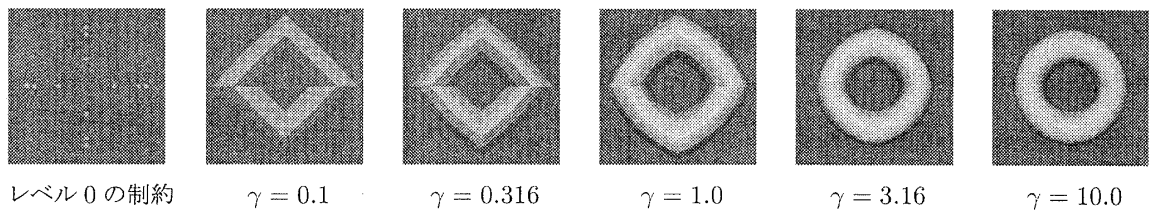


図 4: 解像度レベル 0 の制約と, 滑らかさの異なるレベル 4 のトーラスの形状. γ の値は (39) を参照.

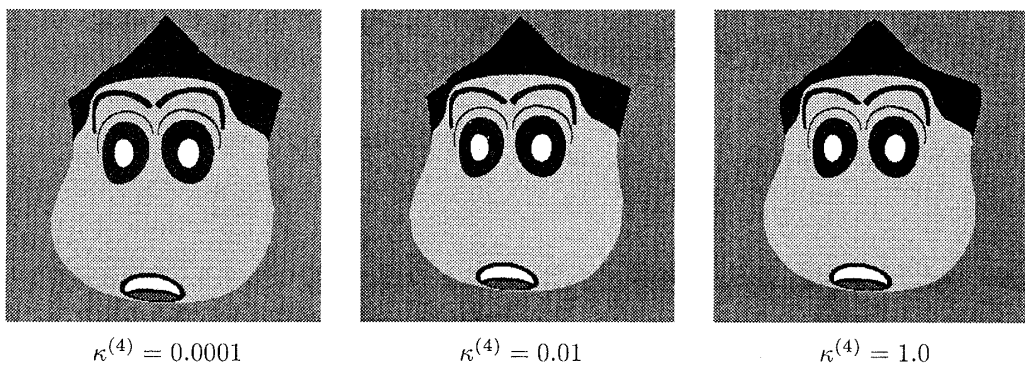


図 5: 解像度レベル 4 の重み $\kappa^{(4)}$ だけを変えたときの髪の変化.