

制約付き不規則メッシュ上での wrinkly 曲面の形成

藤本 忠博^{†,☆} 大野 義夫^{††}

本論文では、補間曲面とフラクタルとを融合した「wrinkly 曲面 (wrinkly surface)」を不規則メッシュ上に形成する方法に関する提案を行う。Wrinkly 曲面は、通常の補間曲面と同様、制御点からなる制御メッシュ上に形成されるパラメトリック曲面であるが、その制御点の補間手続きに「反復交差変換 (Iterated Shuffle Transformation: *IST*)」と呼ぶ変換形式を利用することで、空間方向の局所性とスケール方向の自己相似性とを融合した「空間/スケール方向の局所的類似性」の性質を曲面上に生み出す。Wrinkly 曲面は、各制御点の 3 次元位置や属性値 (高さ値, 色, 透過率等) の調整による柔軟な操作性の下で、従来の大域的なフラクタル手法では困難である局所的な形状の制御を可能とする。著者らは先に、wrinkly 曲面を正方メッシュ上に形成する方法について提案した。本論文では、2 次元の反復交差変換を「グローバル軸の 2 系統化」という制約条件を満たす不規則メッシュ上で適用可能なものに拡張し、その制約付き不規則メッシュ上に wrinkly 曲面を形成する方法について提案を行う。例としていくつかの代表的な不規則メッシュ構造に対する wrinkly 曲面を取りあげたのち、具体的な形状モデルの作成例を通して、wrinkly 曲面が形状モデリングの一手法として有効であることを示す。

Generating Wrinkly Surfaces on Constrained Irregular Mesh

TADAIRO FUJIMOTO^{†,☆} and YOSHIO OHNO^{††}

In this paper, we propose the method of generating a “wrinkly surface” on an irregular mesh. A wrinkly surface is a unification of an interpolation surface and a fractal surface, and it is a parametric surface which is generated on a mesh of control points as a usual interpolation surface is. The interpolation procedure called “Iterated Shuffle Transformation: *IST*” applied to control points gives the property of “local resemblance in space/scale direction”. This means that the surface has locality in the space direction and self-similarity in the scale direction. By changing the control point positions and their attributes such as height values, colors, or transparency, we can modify the shape locally; this is difficult for usual global fractal methods. We presented a method of generating a wrinkly surface on a regular mesh before. In this paper, we improve two-dimensional Iterated Shuffle Transformation to be applied on an irregular mesh which is constrained by the condition “two-grouping of global axes”, and propose the method of generating a wrinkly surface on the constrained irregular mesh. After giving several wrinkly surfaces on typical irregular meshes as examples, we show the excellent ability of wrinkly surface for shape modeling through making some shape models actually.

1. 序 論

著者らは文献 14) で「手続き的モデリング」^{2),3)}と「制御点補間」¹⁾とを融合した「手続き的補間 (procedural interpolation)」の概念を提唱し、その具体例として「wrinkly 曲面 (wrinkly surface)」という曲面形式の提案を行った。Wrinkly 曲面は、従来

の補間曲面^{☆☆}と同様に、3 次元空間上の制御点からなる制御メッシュ上にパラメータ (u, v) に関するパラメトリック曲面として形成される。各制御点は 3 次元位置座標値と変位のための高さ値を持ち^{☆☆☆}、それらが「反復交差変換」^{13),14)}という座標変換手続きにより区分的に補間された結果、displacement mapping の変位による凹凸を持つ曲面として wrinkly 曲面が形成さ

† 慶應義塾大学大学院理工学研究科
Graduate School of Science and Technology, Keio University

☆ 現在、岩手大学工学部
Presently with Faculty of Engineering, Iwate University

†† 慶應義塾大学理工学部
Faculty of Science and Technology, Keio University

☆☆ 本論文では、Bézier 曲面や B-spline 曲面等、制御点の滑らかな補間により生成される曲面を総称して「補間曲面」と呼ぶこととする。文献 14) では「スプライン曲面」を総称としたが、狭義のスプライン曲面との誤解を避けるため、本論文では「補間曲面」とした。

☆☆☆ そのほかにも、色、透過率等の任意の属性値を持たせることが可能である。

れる^{☆1}。「反復交差変換」は、コントロール集合およびIFS法^{5),6),9)}に基づいて著者らが考案したものであり、空間方向の局所性とスケール方向の自己相似性とを融合した「空間/スケール方向の局所的類似性」という特性を生み出す。その結果として、wrinkly 曲面は補間曲面とフラクタルとを融合した曲面形式であると見なすことができる。

著者らが文献14)で提案したwrinkly 曲面は、1次元のwrinkly 曲線を直積の形で2次元に拡張したものであり、制御メッシュが正方メッシュとなる場合に限定されていた。本論文では、これを不規則メッシュの場合に拡張し、不規則メッシュ上でのwrinkly 曲面の形成方法について提案を行う。1次元の反復交差変換を2軸方向に独立に適用するだけでよい正方メッシュの場合と異なり、不規則メッシュの場合には、2次元上での反復交差変換に不都合が生じないようにするため、メッシュに対して「グローバル軸の2系統化」(3.1節)という制約条件が必要となる。すなわち、本論文の手法が適用できるのは、この制約条件を満たす制約付き不規則メッシュである。

本論文では、以下、2章で正方メッシュ上でのwrinkly 曲面の形成について簡単に説明し、3章で制約付き不規則メッシュ上でのwrinkly 曲面の形成に関する提案、4章で事例の提示、5章で結論を述べる。

2. 正方メッシュ上でのwrinkly 曲面

本章では、1次元軸上での反復交差変換、および、正方メッシュ上でのwrinkly 曲面に関する説明を行う。

2.1 反復交差変換

反復交差変換 (Iterated Shuffle Transformation: IST) は、図1に示す方法で、2つの座標軸 u_s と u_d の間に、

$$u_d = \mathcal{F}_{IST}(u_s) \tag{1}$$

という対応関係を構築する座標変換手続きである (図1中のはじめの d^0 軸を u_s 座標軸、 K 回の反復後の d^K 軸を u_d 座標軸とする)。 \mathcal{F}_{IST} は u_s と u_d の座標値を 1:1 対応の関係で対応付ける。

IST はコントロール集合およびIFS法^{5),6),9)}に基づいて著者らが考案した^{13),14)}のものであるが、a) 縮小変換ではなく小区間の細分割により座標軸上の座標値を混合する、b) 座標変換が区分的かつ局所的である、c) 任意の反復回数の下で定義される、という点に関してコントロール集合やIFS法とは異なる。

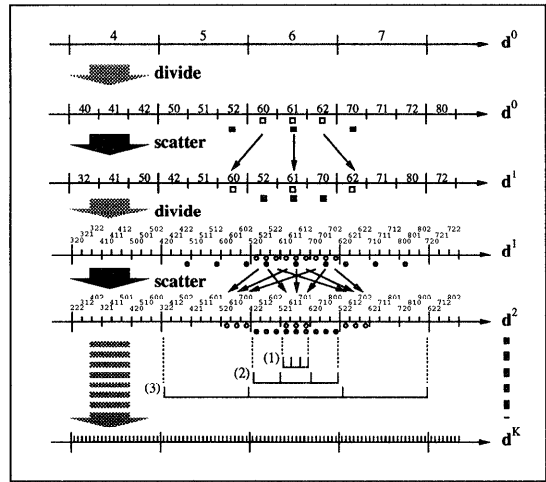


図1 反復交差変換：はじめの座標軸 d^0 をいくつかのベース区間 A_i に区切り (一番上の d^0 軸上に記された 4, 5, 6, 7 ($= i$) は各ベース区間に付けた番号)、各ベース区間 A_i に交差区間数 M_i を与える (この例ではすべての i に関して $M_i = 1$)。あらかじめ決めた反復回数 K になるまで、反復ごとに座標軸が $d^0 \rightarrow d^1 \rightarrow d^2 \rightarrow \dots \rightarrow d^K$ のように変換され、軸上に細分割された小区間がつくられていく (1 回の反復ごとにベース区間 A_i 内の小区間は $2M_i + 1$ 分割される)。小区間を表すための番号付けは、 $6 \rightarrow \{60, 61, 62\} \rightarrow \{\{600, 601, 602\}, \{610, 611, 612\}, \{620, 621, 622\}\} \rightarrow \dots$ という具合に、分割に従って下位の方向に n 進数で表記していくものとする (この例では $n = 2M_i + 1 = 2 \cdot 1 + 1 = 3$)。図中の白黒の四角および丸印は細分割された各小区間が変換によりどこに移るかを表す。 d^0 軸と d^K 軸の間の座標値の対応関係を与えるものとして反復交差変換 \mathcal{F}_{IST} が定義される。

Fig. 1 Iterated Shuffle Transformation.

図1にみられるように、 IST によって構築される座標軸上の小区間の不連続的な並び方は、空間方向とスケール方向^{☆2}の双方に関するわずかな移動の下で類似した並び方を維持する。たとえば、図1中に示した d^2 軸上の小区間番号の並び方について、(1)、(2)、(3)の範囲ごとに3桁の小区間番号の最上桁の数字に注目すると、各範囲内で3つに区切られたそれぞれの区間は、互いにその数字が1ずつずれたものとなっている^{☆3}。この関係は任意の反復回数 k に対する d^k 軸上で成り立つ^{☆4}。 IST が座標軸上につくり出すこの性質を「空間/スケール方向の局所的類似性」と呼ぶこ

☆2 「空間方向」とは「縦、横、高さ」のような「視点」の移動に関する方向性であり、「スケール方向」とは「拡大、縮小」という「視野」の移動に関する方向性である。

☆3 他の桁についても、それぞれに規則性 (ただし、最上桁の規則性とは異なる) が維持されている。

☆4 図1はすべてのベース区間の交差区間数を1としたケースであるが、一般にすべてのベース区間の交差区間数を共通とした場合、任意の交差区間数について同様の性質が成り立つ。

☆1 曲面が凹凸を持つことから、曲面の名前に「しわになりやすい」という意味を持つ「wrinkly」という言葉を用いた。

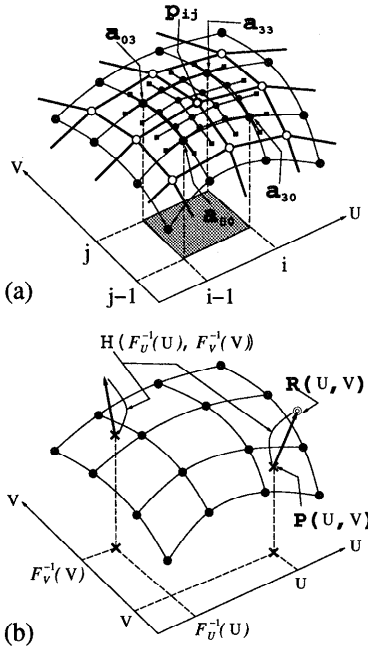


図2 正方メッシュ上の wrinkly 曲面：(a) 白丸は wrinkly 制御点，黒丸は Bézier 制御点（大黒丸は4隅の制御点），黒四角は \mathbf{p}_{ij} に対する Bézier パッチに隣接する周囲の Bézier パッチの制御点の一部を表す。(b) パラメータ (U, V) に対して，基準曲面上の点 $\mathbf{P}(U, V)$ から法線方向に高さ $H(\mathcal{F}_U^{-1}(U), \mathcal{F}_V^{-1}(V))$ だけ持ち上げた点を wrinkly 曲面上の点 $\mathbf{R}(U, V)$ とする。

Fig. 2 Wrinkly surface on regular mesh.

ととする。これは空間方向の局所性とスケール方向の自己相似性とを融合した概念である。

IST を適用する座標軸のタイプとしては，軸の両端が開放されている「並進軸」と両端が閉じている「巡回軸」の2種類が考えられる。

なお，詳細は文献13), 14)を参照のこと。

2.2 正方メッシュ上での wrinkly 曲面の定義

Wrinkly 曲面は「wrinkly 制御点」と呼ぶ制御点からなる制御メッシュ上に形成される。Wrinkly 制御点は3次元空間上での位置座標値と displacement mapping の変位のための高さ値を持つ。本節では，制御メッシュが正方メッシュの場合について説明する(図2, 詳細は文献14)を参照のこと)。

はじめに wrinkly 制御点の位置座標値から基準曲面 $\mathbf{P}(U, V)$ を定義する*。図2(a)に示すように，wrinkly 制御点 \mathbf{p}_{ij} に対してグローバルなパラメータ U, V に関するパラメータ領域 $A_{ij} = \{(U, V) | U \in$

$[i-1, i], V \in [j-1, j]\}$ (図2(a)中の色付きの4角領域)を割り当て，その領域上でローカルなパラメータ u, v ($0 \leq u, v \leq 1$) に対する3次 Bézier パッチ $\mathbf{P}_{ij}(u, v)$ を定義する。これは， \mathbf{p}_{ij} を取り囲む4個の格子領域**のそれぞれの重心位置として4隅の Bézier 制御点 $\mathbf{a}_{00}, \mathbf{a}_{30}, \mathbf{a}_{03}, \mathbf{a}_{33}$ を求め，さらに隣接する周囲の Bézier パッチと G^1 連続となるように他の12個の Bézier 制御点を求めることで決定される(図2(a), Bézier 制御点の具体的な計算方法については3.4節)。基準曲面 $\mathbf{P}(U, V)$ 全体は，パラメータ領域 A_{ij} ごとに区分的に定義される。

$$\mathbf{P}(U, V) = \{\mathbf{P}_{ij}(u, v)\} \tag{2}$$

続いて，基準曲面の定義と同様の Bézier 補間により，wrinkly 制御点の高さ値から区分的に高さ関数 $H(U, V)$ を定義する。

$$H(U, V) = \{H_{ij}(u, v)\} \tag{3}$$

Wrinkly 曲面 $\mathbf{R}(U, V)$ は，図2(b)に示すように，グローバルなパラメータ軸 U, V のそれぞれに対して反復交差変換 $\mathcal{F}_U, \mathcal{F}_V$ を適用し，式(3)の高さ関数による displacement mapping を行うことで生成される。 $\mathcal{F}_U, \mathcal{F}_V$ の定義については， U, V 軸上で上記のパラメータ領域 A_{ij} で区切られる区間，すなわち， U 軸上の $[i-1, i]$ および V 軸上の $[j-1, j]$ (i, j は整数)を各軸上でのベース区間とし，各軸ごとに全ベース区間に共通の交差区間数 M_U, M_V を与える。 M_U, M_V は U, V それぞれの軸方向における wrinkly 制御点の曲面に対する局所的な影響範囲に関する尺度という意味で，従来の補間曲面の次数に相当するものととらえることができる。もちろん， M_U と M_V を共通の値としてもよい(それが自然であるとも考えられる)。また， \mathcal{F}_U と \mathcal{F}_V の反復回数は，曲面に「適応的サンプリング」(4章，詳細は文献14))を適用する際に決定する。Wrinkly 曲面上の点 $\mathbf{R}(U, V)$ は次式で求められる。

$$\mathbf{R}(U, V) = \mathbf{P}(U, V) + H(\mathcal{F}_U^{-1}(U), \mathcal{F}_V^{-1}(V))\mathbf{N}(U, V) \tag{4}$$

ここで， $\mathbf{N}(U, V)$ は基準曲面上の単位法線ベクトルを表す。

図2(b)に示すように，wrinkly 曲面の変位の与え方はフラクタルの典型例である中点変位法^{2)~4), 7), 10), 11)}のものとは異なり，基準曲面上の各点が属性として持つ高さ値を IST に基づいて面方向に移動させること

* 本論文では，wrinkly 曲面のパラメータ変数に関して，グローバルなものは大文字 (U, V) で，ローカルなものは小文字 (u, v) で表記することとする。

** 格子領域とはメッシュ上の格子点 (wrinkly 制御点) を頂点とする多角形領域を意味する。

で変位を決定する。これにより、wrinkly 曲面には空間方向の局所性とスケール方向の自己相似性とを合わせた「空間/スケール方向の局所的類似性」が実現される。また、高さ値以外にも、色、透過率等のさまざまな属性値を wrinkly 制御点に与え、*IST* を適用することが可能である。

3. 制約付き不規則メッシュ上での wrinkly 曲面

本章では、反復交差変換を 2 次元上に拡張し、wrinkly 曲面を制約付き不規則メッシュ上に形成する方法について提案する。

3.1 不規則メッシュに対する制約条件

Wrinkly 曲面の定義においては、グローバルなパラメータ軸上での *IST* の適用が最も重要な役割を果たす。正方メッシュの場合にはグローバルな U, V の 2 軸方向にそれぞれ独立に *IST* を適用するだけでよいのに対し、不規則メッシュに対する *IST* の適用を考えた場合には、単純に 1 次元の *IST* を直積の形で 2 次元に拡張することはできない。この 1 次元から 2 次元への *IST* の拡張を考えるにあたり、本論文で採用した基本方針は、

「 U, V 軸方向の変換を逐次的に 2 段階で行う。」
 … (方針☆)

というものである。すなわち、何らかの方法でグローバルな U, V 軸群を不規則メッシュ上に配置し、はじめに U 軸方向に対して *IST* を適用した後、 V 軸方向に対して適用を行う（適用の順序が逆の場合も可）、という方針である。そして、この基本方針の下では、以下に述べる制約条件を不規則メッシュに与える必要が生じる。

まず、不規則メッシュ上での wrinkly 曲面の定義方法として、正方メッシュの場合と同様に、各 wrinkly 制御点ごとにローカルなパラメータ u, v に関する 4 角パッチを定義し、それらを区分的につなぐことで曲面全体を構成することとする。このパッチごとのローカル軸 u, v を連結したものをグローバル軸とすると、結果として曲面全体にグローバル軸群が配置されることになる (図 3)。上記の (方針☆) の下で拡張された 2 次元の *IST* を妥当なものとするためには、このグローバル軸群の配置に対してある制約条件、すなわち、以下に述べる「グローバル軸の 2 系統化」の条件が必要となる。

曲面全体は複数のグローバル軸で覆われることになるが、このうち互いに平行な部分を持つ軸どうしは、その部分では同じ方向性を持ち、共通の *IST* を適用

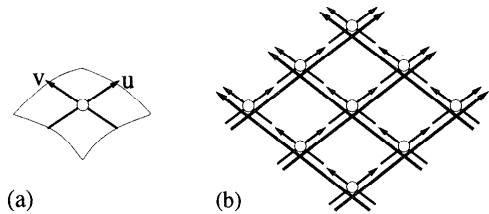


図 3 ローカル軸とグローバル軸：(a) Wrinkly 制御点 (白丸) に付随するローカル u, v 軸 (矢印) に対して 4 角パッチが定義される。(b) ローカル軸 (短い矢印) を連結したものをグローバル軸 (長い矢印) とする。グローバル軸群は wrinkly 制御点を連結するものと解釈でき、曲面全体を覆う制御メッシュとなる。

Fig. 3 Local coordinate axis and global coordinate axis.

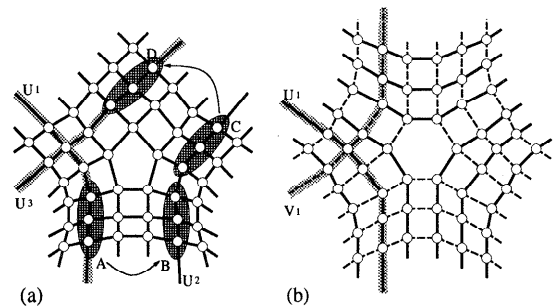


図 4 グローバル軸の系統分け：Wrinkly 制御点 (白丸) をグローバル軸が通過する。(a) 1 系統の例。軸 U_1 と U_3 は平行関係を持たないように見えるが、「軸 U_1 の A 部分と軸 U_2 の B 部分が平行 → 軸 U_2 の C 部分と軸 U_3 の D 部分が平行」により、軸 U_2 を仲介して軸 U_1 と U_3 は平行関係を持つことがわかる。(b) 2 系統の例。軸 U_1 と V_1 は平行関係を持ち得ない。全グローバル軸が「 U 系統」(実線)か「 V 系統」(点線)のいずれかに所属する。

Fig. 4 Grouping global coordinate axes.

したい。そこで、一部分でも平行関係にある軸どうしは共通の「系統」に属するものとし、すべてのグローバル軸の系統分けを行う。すると、図 4 の例に示すように、すべての wrinkly 制御点グローバル軸を通じて連結し孤立しているものはないという条件の下で、グローバル軸群のあらゆる配置のしかたを考えた場合に、それらは、すべての軸が 1 系統に属してしまう [すなわち、任意の 2 軸が直接、あるいは、他軸との平行部分を仲介した平行部分を持つ (図 4(a)) 配置、2 系統に属する [すなわち、前者 (1 系統) のような平行部分を互いに持たない「 U 系統」と「 V 系統」のどちらか一方に各軸が所属する (図 4(b)) 配置、のいずれかの配置をとることが分かる (証明は付録を参照)。

ここで、上記の (方針☆) の下での 2 次元の *IST* の適用を考えると、図 5 (a) に示すように、グローバル軸群の配置が 1 系統となる場合には座標変換に不都合が生じることが分かる。一方、2 系統となる場合は、

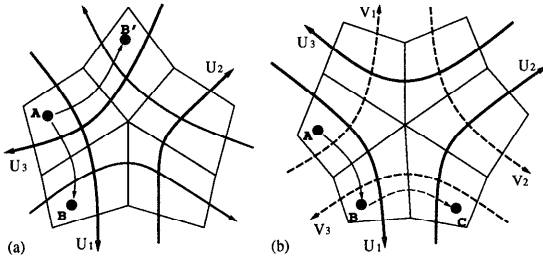


図5 2次元反復交差変換の妥当性：(a) 1系統の例。2段階の変換を行う場合、仮に U_1 軸方向を1回目の変換方向として選択すると、 U_1 軸に平行な U_2 軸、そして、 U_2 軸に平行な U_3 軸も1回目として同時に選択される。その結果、 U_1 、 U_3 軸を持つパッチ上の点 A はどちらの軸方向に1回目の変換を行うべきか(点 B と B' のいずれか) 決定できなくなる。(b) 2系統の例。同系統に属する2軸を持つパッチが存在しないため、(a) のような不都合が生じない ($A \xrightarrow{U \text{ 軸方向}} B \xrightarrow{V \text{ 軸方向}} C$)。

Fig. 5 Validity of two-dimensional Iterated Shuffle Transformation.

各グローバル軸は「 U 系統」と「 V 系統」のいずれかにはっきりと分類できるため、 U 軸方向、 V 軸方向の2段階の変換が妥当に行われる(図5(b))。

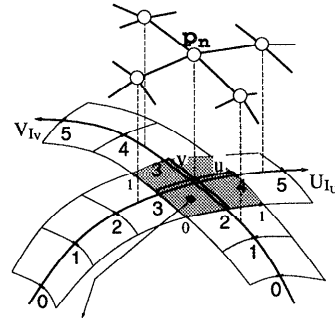
以上のことから、本論文では、「グローバル軸の2系統化」の制約条件を満たす制約付き不規則メッシュのみを対象とし、 U 、 V 軸方向に逐次的に IST を適用することで2次元上への IST の拡張を実現することとする。なお、以下、特に断りのない場合には、この制約付き不規則メッシュという意味で「不規則メッシュ」という用語を用いることとする。

3.2 2次元反復交差変換

3.2.1 不規則メッシュ上への一般化のために検討すべき点

正方メッシュは前節の2系統化の条件を満たす一例である。正方メッシュ上の各(wrinkly 制御点に対して定義される)4角パッチを通過するグローバル軸を考えた場合、同じ系統に属するすべての軸が全体を通じて完全に平行であるため、各系統ごとに原点と方向を一致させ、1つの軸で代表させることが可能である。そのため、互いに独立した U 、 V の2軸のみで曲面上の任意の点のグローバルパラメータ値を記述でき、その2軸に独立に IST を適用することで2次元上の IST が実現可能となる(図8(a))。

一方、不規則メッシュの場合には、同じ系統に属するグローバル軸であっても部分的な平行関係しか持たないため、同系統内の軸を原点と方向を統一した1つの軸で代表させることはできない(図8(b))。そこで、2次元上の IST を実現するためには、それぞれのグローバル軸どうしの位置関係に基づき、各軸ごと



$$F_{LG} \begin{bmatrix} t = (n, (u, v)) \\ T = ((I_U, U), (I_V, V)) \end{bmatrix} F_{LG}^{-1}$$

図6 ローカル軸のグローバル軸への埋め込み：白丸は wrinkly 制御点を表す。 U_{I_U} 、 V_{I_V} はそれぞれ軸番号 I_U 、 I_V のグローバル U 、 V 軸である。色付きの四角領域は wrinkly 制御点 p_n に割り当てられたパラメータ領域(領域内の矢印線でローカル u 、 v 軸を表す)であり、その内部の点はローカル座標値 t およびグローバル座標値 T で特定化される。

Fig. 6 Imbedding of local coordinate axis to global coordinate axis.

のパラメータ値を何らかの方法で対応付ける必要がある(3.2.3 項)。また、グローバル軸は IST の適用対象として重要な役割を果たすものではあるが、3.3 節で述べるような4角パッチのつなぎ合わせによる wrinkly 曲面の形状そのものの定義方法の下で、不規則なつなぎを持つメッシュを対象として考えた場合には、グローバル軸上でのパラメータ値を直接扱うより、各パッチ内のローカル値を変換値とする、

$$\begin{matrix} \text{(パッチ 1, パッチ 1 内での } (u, v) \text{)} \\ \downarrow \text{2次元 } IST \\ \text{(パッチ 2, パッチ 2 内での } (u, v) \text{)} \end{matrix} \dots \text{(形式★)}$$

という形式の2次元 IST を実現したほうが都合がよい。この場合、それぞれのローカル軸とグローバル軸のパラメータ値の対応関係をいかに構築するかが重要となる(3.2.2 項)。

以上をもとに、不規則メッシュ上への2次元 IST の一般化について、本論文で提案する具体的な実現方法を以下に述べる。

3.2.2 ローカル値とグローバル値の対応

各 wrinkly 制御点ごとのローカル u 、 v 軸のパラメータ値 ($0 \leq u, v \leq 1$) を、図6のようにしてグローバル軸上のパラメータ値に埋め込む。まず、各 wrinkly 制御点の識別番号を n 、その制御点に付随するローカル軸を u 、 v とするとき、このローカル座標値 t を以下のように表記する。

$$t = (n, (u, v)) \quad (0 \leq u, v \leq 1) \quad (5)$$

また、各グローバル軸の識別番号を I 、その軸上での

パラメータ値を U とするとき、そのグローバル座標値を (I, U) と表記する。すると、曲面上の任意の点のグローバル座標値 \mathbf{T} は、その点を含む 4 角パッチを通過するグローバル軸の番号を I_U, I_V 、それらの上でのパラメータ値を U, V とするとき、

$$\mathbf{T} = ((I_U, U), (I_V, V)) \tag{6}$$

と表すことができる。ここで、各グローバル軸上でのパラメータ値に関して、図 6 に示すように、各グローバル軸ごとに、自分に埋め込まれるローカル軸が各々グローバルなパラメータ長 1 の区間に対応するように（すなわち、正方メッシュの場合と同様に）グローバル値を割り振る。これにより、式 (5) と式 (6) の間に、次に示す「ローカル値 \leftrightarrow グローバル値」の変換式が構築できる。

$$\mathbf{T} = FLG(\mathbf{t}), \quad \mathbf{t} = FLG^{-1}(\mathbf{T}) \tag{7}$$

以上のローカル軸とグローバル軸の対応関係は、具体的には、図 7 に示すアルゴリズムにより、与えられた不規則メッシュ情報からグローバル軸群を構築することで得られる。図 7 のアルゴリズムでは、各々の wrinkly 制御点がどの制御点と連結しているかというリンク情報として不規則メッシュが与えられ、それをもとにして、 U, V 系統のグローバル軸群を構築する。その処理過程で内部データとして保持された情報から式 (7) が実現される。

3.2.3 平行なグローバル軸間でのパラメータ値の対応

グローバル軸間のパラメータ値の対応付けについては、軸番号 I, J の 2 つのグローバル軸が平行な部分を持つとき、その平行部分での双方の上でのパラメータ値 U_I, U_J を対応付ける以下の変換式を定義する (図 8 (b))。

$$\begin{aligned} U_J &= G_{PAR}(I, J, U_I) \quad \text{または} \\ U_I &= G_{PAR}(J, I, U_J) \end{aligned} \tag{8}$$

式 (8) の変換も図 7 のアルゴリズムで構築される情報をもとに行われる。

3.2.4 2次元反復交差変換の定義

不規則メッシュの場合にも、正方メッシュの場合と同様に、グローバル軸上で各ローカル軸の埋め込みに対応するパラメータ長 1 の区間を IST のベース区間とし、 U, V 系統ごとに全ベース区間に共通の交差区間数 M_U, M_V を与え、反復交差変換 $\mathcal{F}_U, \mathcal{F}_V$ を適用する (図 8 (b))。

ここで 1 つの問題が生じる。それは、不規則メッシュのグローバル軸群が 2 系統化の条件を満たしたとしても、一般に、 U, V 系統に IST を適用する順序の違

いによって変換結果が変わってしまう、ということである。正方メッシュの場合には、図 8 (a) のように、グローバル値が $A(U, V)$ から $C(U', V')$ へ変換される時、次の関係が成り立つ。

$$(U', V') = (\mathcal{F}_U(U), \mathcal{F}_V(V)) \tag{9}$$

すなわち、先にも述べたように、 U, V 軸上での変換がそれぞれ独立に行われるため、その順序が変換結果に影響することはない。一方、不規則メッシュの場合には、図 8 (b) のように、その変換順序により、 $A \xrightarrow{U\text{方向}} B \xrightarrow{V\text{方向}} C$ 、あるいは、 $A \xrightarrow{V\text{方向}} B' \xrightarrow{U\text{方向}} C'$ という経路をたどり、点 A が異なる点 C, C' に移されてしまう。

上記の問題は、3.1 節で述べた (方針☆) の下で IST を 2 次元に拡張することを前提とした場合、避けられない問題である。したがって、本論文では、1 つの解決策として、「(高さ関数等の、2 次元 IST を適用しようとする) 目的のパラメータ関数に対して、「 $U \rightarrow V$ 」と「 $V \rightarrow U$ 」の両方の順序で適用を行い、それぞれから得られる関数値の加重和を計算する」という方法 (詳細は本章で後述、式 (12)) を採用することとする。

不規則メッシュ上でのグローバル値 (式 (6)) に関する 2 次元 IST を、「 $U \rightarrow V$ 」および「 $V \rightarrow U$ 」の変換順序のそれぞれについて、次式で記述することにする (式中の $\mathbf{T}, \mathbf{T}', \mathbf{T}''$ ☆はグローバル値とする)。

$$\begin{aligned} \mathbf{T}' &= \mathcal{F}_{U \rightarrow V}^M(\mathbf{T}) \quad \text{および} \\ \mathbf{T}'' &= \mathcal{F}_{V \rightarrow U}^M(\mathbf{T}) \end{aligned} \tag{10}$$

式 (10) は式 (9) を不規則メッシュへ一般化した表記であり、正方メッシュでは \mathbf{T}' と \mathbf{T}'' が一致する。ここで、3.2.1 項の (形式★) に従ってローカル値 (式 (5)) を対象とする 2 次元 IST を実現することを考えた場合、式 (10) および式 (7) より、ローカル値 \mathbf{t}, \mathbf{t}' に関する 2 次元 IST として次式が記述できる。

$$\begin{aligned} \mathbf{t}' &= \mathcal{F}_{U \rightarrow V}^{M, LOC}(\mathbf{t}) \\ &= FLG^{-1}(\mathcal{F}_{U \rightarrow V}^M(FLG(\mathbf{t}))) \\ &\quad (\mathcal{F}_{V \rightarrow U}^{M, LOC} \text{も同様。}) \end{aligned} \tag{11}$$

式 (10) および (11) は、1 次元の IST である式 (1) と同様に、パラメータ空間上での 1:1 対応の変換を実現している☆☆。

不規則メッシュ上で定義される任意のパラメータ関

☆ 図 8 (b) の場合には $\mathbf{T}, \mathbf{T}', \mathbf{T}''$ が各々点 A, C, C' のグローバル値に対応する。

☆☆ 3.1 節の (方針☆) により、式 (10), (11) では U, V それぞれの軸方向についての変換が逐次的に行われる。そのそれぞれがパラメータ空間上で 1:1 対応の変換であることは明らかである。よって、式 (10), (11) は 1:1 対応の変換であるといえる。

下記のアルゴリズムにより、不規則メッシュ上の各 wrinkly 制御点 Pon に関して次の情報を求める。

Pon.prv_u, nxt_u, prv_v, nxt_v ... 自分のローカルな $-u$, $+u$, $-v$, $+v$ 方向の制御点 (「リンク情報」)。

Pon.crd_U, crd_V ... 自分を通過するグローバル U , V 軸 (「グローバル軸情報」)。

また、グローバル U , V 軸の総数 G_U_n , G_V_n , および、各グローバル軸が通過する先頭、後尾制御点の情報 G_Pon_U , G_Pon_V (「末端制御点情報」) を求める。

```

/* 初期状態. */
集合 Set_Pon = { 不規則メッシュ上の全 wrinkly 制御点 (リンク情報あり). }

グローバル  $U$ ,  $V$  軸の総数の初期化.    $G\_U\_n = 0$    $G\_V\_n = 0$ 

グローバル  $U$ ,  $V$  軸を構築するためのスタック stack_U, stack_V に Set_Pon 中の先頭の制御点を置く.

/* グローバル  $U$ ,  $V$  軸の構築. */
while (stack_U, stack_V がともに「空」になるまで繰り返す.) {
  /* グローバル  $U$  軸の構築. */
  while (stack_U が「空」になるまで繰り返す.) {
    探索を開始する制御点を取り出す.  stack_U  $\Rightarrow$  str_Pon
    if (str_Pon がすでにグローバル  $U$  軸を持つ.) { stack_U から次の取り出し. }
    新たなグローバル  $U$  軸 cur_Cord_U の初期化 ( $G\_U\_n$  番目のグローバル  $U$  軸).
    /*  $U$  軸の正方向の探索. */
    cur_Pon  $\leftarrow$  str_Pon      nxt_Pon  $\leftarrow$  cur_Pon.nxt_u
    while (nxt_Pon が「空」または「str_Pon に戻る」まで繰り返す.) {
      if (nxt_Pon がまだグローバル  $V$  軸を持たない.) { stack_V  $\leftarrow$  nxt_Pon } ... (処理☆)
      nxt_Pon.prv_u が cur_Pon と一致するように
      nxt_Pon のリンク情報を入れ換える. ... (処理 1)
      cur_Pon.crd_U  $\leftarrow$  現在のグローバル  $U$  軸 cur_Cord_U を登録. ... (処理 2)
      cur_Pon  $\leftarrow$  nxt_Pon      nxt_Pon  $\leftarrow$  cur_Pon.nxt_u
    }
    /*  $U$  軸の負方向の探索. */
    正方向の探索と同様に行う.
    /* グローバル  $U$  軸の決定. */
    探索結果から,  $G\_Pon\_U[G\_U\_n]$  に先頭, 後尾制御点を登録.  $G\_U\_n++$  ... (処理 3)
  }
  /* グローバル  $V$  軸の構築. */
  グローバル  $U$  軸の構築と同様に行う. ... (処理 4)
}

```

集合 Set_Pon 中の先頭の制御点から隣接する制御点を「イモツル」式に探索していくことで、曲面上の全グローバル軸を構築する。初期状態の集合 Set_Pon 中で与えられる「リンク情報」は各制御点ごとにローカルなものでありグローバルな整合性はなく、(処理 1) 中でグローバルな整合性を満たすように書き換えられる。(処理 2) により各制御点を通過するグローバル軸が「グローバル軸情報」に登録される。(処理 3) による「末端制御点情報」と各制御点の「リンク情報」により、各グローバル軸がどの制御点を通過するのか特定できるようになる。先頭制御点の $u = 0$ (または $v = 0$) を原点として後尾制御点まで制御点ごとにグローバル軸上の値を 1 ずつ増し、その値を各制御点に保持させておくで、図 6 の埋め込みを実現する。(処理☆) および (処理 4) 中の同処理により曲面上の隣り合う制御点が次々とスタックに積まれ、結果として曲面全体の処理が完成する。

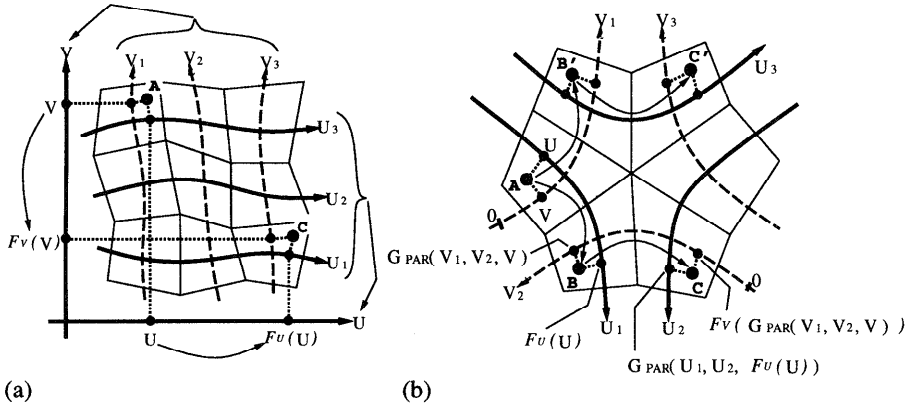
図 7 グローバル軸群を構築するアルゴリズムの疑似コード

Fig. 7 Pseudo code of algorithm for constructing global coordinate axes.

数を $F_A(\mathbf{t})^*$ とすると、上記の「加重和」による方法

では、関数 F_A に手続き \mathcal{P}_{IST}^M を適用して新たな関数 $[\mathcal{P}_{IST}^M(F_A)]$ を構築する。

* 実際には、関数 F_A はローカル値 \mathbf{t} , グローバル値 \mathbf{T} のいずれに関して定義されていてもよい。ここではローカル値に関する関数として考えており、式 (12) にもローカル値に関する変換式 (11) を用いている。



(a) 正方メッシュの場合. U_1, U_2, U_3 軸は U 軸, V_1, V_2, V_3 軸は V 軸によって, それぞれ原点と方向を一致させて代表させることが可能であり, U, V の 2 軸のみに独立に IST を適用するだけでよい. (b) 不規則メッシュの場合. 「 $U \rightarrow V$ 」順序でのグローバル値の変換は $A((U_1, U), (V_1, V)) \xrightarrow{U \text{ 方向}} B((U_1, \mathcal{F}_U(U)), (V_2, G_{PAR}(V_1, V_2, V))) \xrightarrow{V \text{ 方向}} C((U_2, G_{PAR}(U_1, U_2, \mathcal{F}_U(U))), (V_2, \mathcal{F}_V(G_{PAR}(V_1, V_2, V))))$ となる. ここで関数 G_{PAR} は 3.2.3 項で述べた平行なグローバル軸間でのグローバル値の変換を行う関数である (ここでは G_{PAR} に引数として与える各軸の番号を U_1, U_2, V_1, V_2 のように記述した). たとえば, 図中の点 A, B は各々グローバル軸 V_1, V_2 上で V 値を与えられるが, V_1 と V_2 では原点および軸方向が異なるため, それぞれの V 値は同じ値にはならない. よって, $A \xrightarrow{U \text{ 方向}} B$ の変換時には, グローバル軸 U_1 に沿って反復交差変換 \mathcal{F}_U により U 値を変換すると同時に, V 値についても G_{PAR} によって変換を行わなければならない.

図8 2次元反復交差変換.

Fig. 8 Two-dimensional Iterated Shuffle Transformation.

$$\begin{aligned}
 & [P_{IST}^M(\mathcal{F}_A)]((\alpha_0, \alpha_1), t) & (12) \\
 & = \alpha_0 \mathcal{F}_A(\mathcal{F}_{U \rightarrow V}^M \mathcal{L}OC^{-1}(t)) + \alpha_1 \mathcal{F}_A(\mathcal{F}_{V \rightarrow U}^M \mathcal{L}OC^{-1}(t)) \\
 & (\alpha_0 + \alpha_1 = 1, 0 \leq \alpha_0, \alpha_1 \leq 1)
 \end{aligned}$$

図8(b)の場合には, 式(12)により得られる点 A における変換値は, 点 C と C' における関数値を取り込み, それぞれを α_0, α_1 の重み付きで加算したものとなる. この混合パラメータ α_0, α_1 は, 2次元 IST により曲面上の対応する点から関数値を取り込む際の (右回り, 左回りといった回転に関する) 方向性を調整する役割を果たすものと考えられる☆.

3.3 不規則メッシュ上での wrinkly 曲面の定義

3.1 節で触れたように, 不規則メッシュの場合にも, 正方メッシュの場合 (2.2 節) と同様に, 各 wrinkly 制御点に対する 4 角パッチを定義し, それらをつなぎ合わせることで wrinkly 曲面の全体形状を構成する (図9). なお, 3.2.1 項で述べたように, 不規則メッシュを対象とした場合の扱いやすさを考えて, これ以降の (基準曲面等の) 関数式の定義はパッチごとのローカル値 t (式(5)) に関して行うこととする.

まず, 基準曲面 $\mathbf{P}(t)$ は, 各 wrinkly 制御点 \mathbf{p}_n に対して 3 次 Bézier パッチ $\mathbf{P}_n(t)$ を定義し, それらを

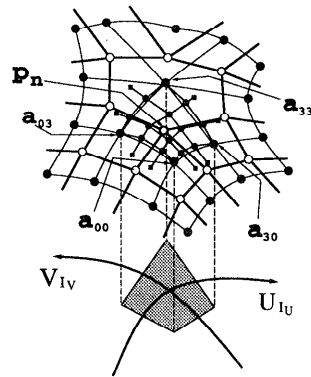


図9 不規則メッシュ上の wrinkly 曲面: 白丸は wrinkly 制御点, 黒丸は Bézier 制御点 (大黒丸は 4 隅の制御点), 黒四角は \mathbf{p}_n に対する Bézier パッチに隣接する周囲の Bézier パッチの制御点の一部を表す. ハッチングされた四角領域は \mathbf{p}_n に割り当てられたパラメータ領域, U_{IU}, V_{IU} はグローバル軸である.

Fig. 9 Wrinkly surface on irregular mesh.

区分的につなぎ合わせて構成する.

$$\mathbf{P}(t) = \{\mathbf{P}_n(t)\} \tag{13}$$

図9では, wrinkly 制御点 \mathbf{p}_n は自分を取り囲む 4 個の格子領域 (1 個の 6 角形と 3 個の 4 角形) の共通の頂点となっている. そこで, まず, それら格子領域のそれぞれの重心位置として Bézier パッチの 4 隅の制御点 $\mathbf{a}_{00}, \mathbf{a}_{30}, \mathbf{a}_{03}, \mathbf{a}_{33}$ を求める. そして, その他

☆ 詳細については今後の検討課題である.

の 12 個の Bézier 制御点は、隣接する周囲の Bézier パッチの制御点 (図 9 中の黒四角) と一直線上に並ぶように求める (詳細は 3.4 節)。

高さ関数 $H(t)$ についても、上記の基準曲面と同様の 3 次 Bézier 補間による方法で、各 wrinkly 制御点の持つ高さ値に基づく区分的な関数として定義する。

$$H(t) = \{H_n(t)\} \quad (14)$$

式 (13), (14) に基づく wrinkly 曲面 $\mathbf{R}(t)$ の定義方法は、正方メッシュの場合と同様の displacement mapping によるものであり、不規則メッシュ上での 2 次元 IST の変換式 (12) を $H(t)$ に適用して以下のように記述できる^{*1}。

$$\mathbf{R}(t) = \mathbf{P}(t) + [P_{IST}^M(H)]((\alpha_0, \alpha_1), t)\mathbf{N}(t) \quad (15)$$

ここで、 $\mathbf{N}(t)$ は基準曲面上の単位法線ベクトルを表す。

3.4 Bézier 制御点の計算方法

図 10 は式 (13) の基準曲面を構成する Bézier パッチの制御点の具体的な計算方法を示す。図 10 (a), (b), (c) および (d) では、中央に置かれた二重白丸の wrinkly 制御点に対する 3 次 Bézier パッチの 16 個の制御点を求めている。図 9 に示すように格子領域の重心位置としてパッチの 4 隅の制御点 (図中の大黒丸) を求めた後、図 10 中に示した内分比^{*2}により他の 12 個の制御点位置を求める。図 10 (a')~(d') は、それぞれ (a)~(d) の wrinkly 制御点のつながり方が実際にメッシュ上に現れる例であり、各々が「グローバル軸の 2 系統化」の条件を満たすメッシュである。(b'), (c'), (d') のようなメッシュの配置 (wrinkly 制御点が $\pm u, \pm v$ の 4 方向のすべてでは連結しない) を許した場合、結果として Bézier パッチが 3 辺形 ((b')) あるいは 2 辺形 ((c'), (d')) に縮退してしまう。(b), (c) および (d) は、このような縮退ケースを考慮したものである^{*3}。なお、この方法では、正方メッシュ以外の任意の不規則メッシュを考えた場合、隣接する Bézier パッチの間で G^1 連続等の連続性が保証されるわけではないが^{*4}、ほとんどのケースでほぼ滑らかとなることが実験的に確かめられた。

メッシュが境界を持つ場合には、図 11 (a) のように境界に沿って仮の wrinkly 制御点を配置してから、上

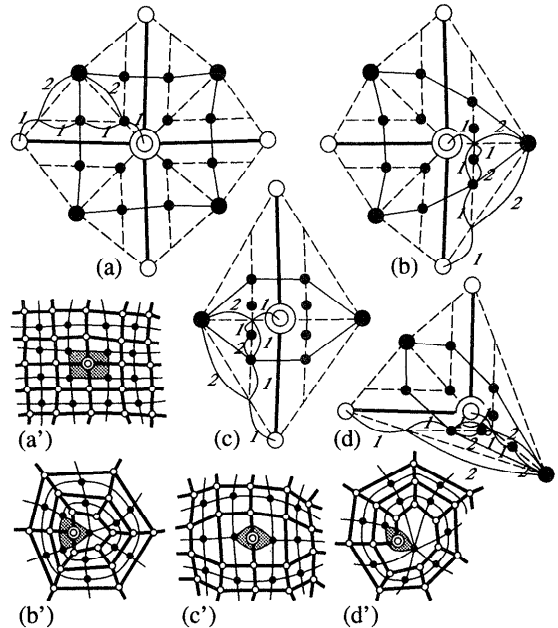


図 10 Bézier パッチの制御点の計算: 白丸は wrinkly 制御点, 黒丸は Bézier パッチの制御点を表す。(a), (b), (c) および (d) は中央の wrinkly 制御点 (二重白丸) に対する Bézier 制御点の計算方法, (a'), (b'), (c') および (d') はその各々が実際のメッシュ中で現れる例 (色の付いた部分が対象となる Bézier パッチ) を示す。(b), (c), (d) の Bézier パッチは縮退し、いくつかの Bézier 制御点が重なることで、3 辺形, 2 辺形となる。

Fig. 10 Calculation of control points of Bézier patch.

記の方法で境界上の wrinkly 制御点に対する Bézier パッチを求める。この仮の wrinkly 制御点に対しては、実際の Bézier パッチを求めることはせず、あくまで境界上の wrinkly 制御点に対する Bézier パッチを求めるためだけに (3 次元位置座標値や高さ値等の) 属性値を与える。

図 11 (b) の左図のメッシュは回転方向のグローバル軸 [巡回軸 (2.1 節)] で囲まれた境界を中央に持つ。このとき、中央に配置された仮の wrinkly 制御点をすべて取り去ることで、右図のように境界が閉じる。この場合、中央の Bézier パッチが 3 辺形に縮退する。

3.5 2 系統化された不規則メッシュの構造

ここで、「グローバル軸の 2 系統化」の条件を満たす不規則メッシュ (2 系統化された不規則メッシュ) が

^{*1} 正方メッシュの場合と同じく、高さ値以外の色、透過率等のさまざまな属性値に対して IST を適用することができる。

^{*2} この比率は、なるべく簡単な整数比により、パッチの形状が各 wrinkly 制御点の位置を適切に反映するものとなるように、経験的に与えたものである。

^{*3} 図 10 (b), (c) および (d) 中に示されている Bézier 制御点の個数は 16 個に満たないが、これは 16 個のうちのいくつかが同じ位置に重なっているものとする。

^{*4} これは、本手法では、Bézier パッチの 4 隅の制御点位置において、周囲のパッチとの接線連続性が保証されないためである。通常の曲面パッチによる形状モデルでは滑らかな曲面形状を目的とするため、 C^m , G^m 等の連続性が重要となるが、高さ関数の displacement mapping に基づく wrinkly 曲面による形状モデルではそれらの連続性はあまり重要ではない。

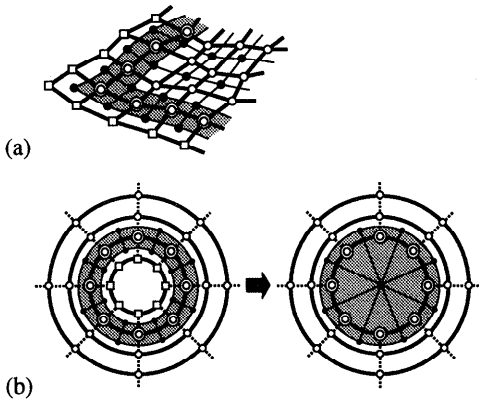


図 11 境界を持つ制御メッシュ：白丸は wrinkly 制御点，二重白丸は境界上の wrinkly 制御点，白四角は仮の wrinkly 制御点，黒丸は Bézier パッチの 4 隅の制御点を表す。(a) 仮の wrinkly 制御点は境界上の Bézier パッチ (色付き) を定義するために配置される。(b) 仮の wrinkly 制御点を取り去ることで中央の境界が閉じる。中央の Bézier パッチ (色付き) が 3 辺形に縮退する。

Fig. 11 Control mesh with boundaries.

いかなる構造を持ちうるのか考えてみたい。

2 系統化された不規則メッシュにみられる代表的な構造は「 N 方向ネット」と呼ぶものである。図 12 (a-1,2,3), (b-1,2,3) および図 13 に示すように, U, V 系統それぞれのグローバル軸群が N 方向 ($N \geq 2$) の方向性を持って交わる構造を持つ。また, 図 12 (c-1,2,3) (図 11 (b) の右図と同種の構造) は「環状ネット」と呼び, その中心部分に放射方向のグローバル軸 [並進軸 (2.1 節)] 群の端点が集申し, 回転方向のグローバル軸 [巡回軸 (2.1 節)] 群がその周囲を囲む構造を持つ。たいていの 2 系統化された不規則メッシュは, 主にこれらをつなぎ合わせることで構成されるものと考えられる*。

図 12, 図 13 中の“wrinkly 曲面”は, 各 wrinkly 制御点に高さ値の他に色 (RGB) 値を与え, 高さ値と色値の双方に IST を適用して生成したものである。一方, 各図中の“補間曲面”は IST を適用しなかった場合である。特に曲面上の色を観察した場合, 補間曲面では各 wrinkly 制御点に与えた色値に従う連続的な変化がみられるのに対し, wrinkly 曲面では「空間/スケール方向の局所的類似性」(2.1, 2.2 節, 文献 14)) を表す不連続的な色の変化がみられる。なお, 各図中の制御メッシュは, 格子点上に wrinkly 制御点が位置し, 先の丸いトゲで高さ値分の法線方向への持ち上げを表す (一番外側の周は仮の制御点)。

4. 実 例

図 14 は不規則メッシュ上の wrinkly 曲面を用いて作成した形状モデルの例である。交差区間数は U, V 系統ともに $M_U = M_V = 1$ とし, 式 (15) 中の混合パラメータは $\alpha_0 = \alpha_1 = 0.5$ とした。また, レンダリングには Z バッファ法を用い, wrinkly 曲面のポリゴン化のための方法には「適応的サンプリング」¹⁴⁾を用いた** (図 12, 図 13 の wrinkly 曲面もこれらの条件で作成した)。なお, 使用した計算機は SGI OCTANE (R10000 (175 MHz×2), 384 MByte) であり, 特に図 14 (c) の画像は計算時間: 32 分 16 秒 (ポリゴン生成 12 分 26 秒, レンダリング 19 分 50 秒) を要した。

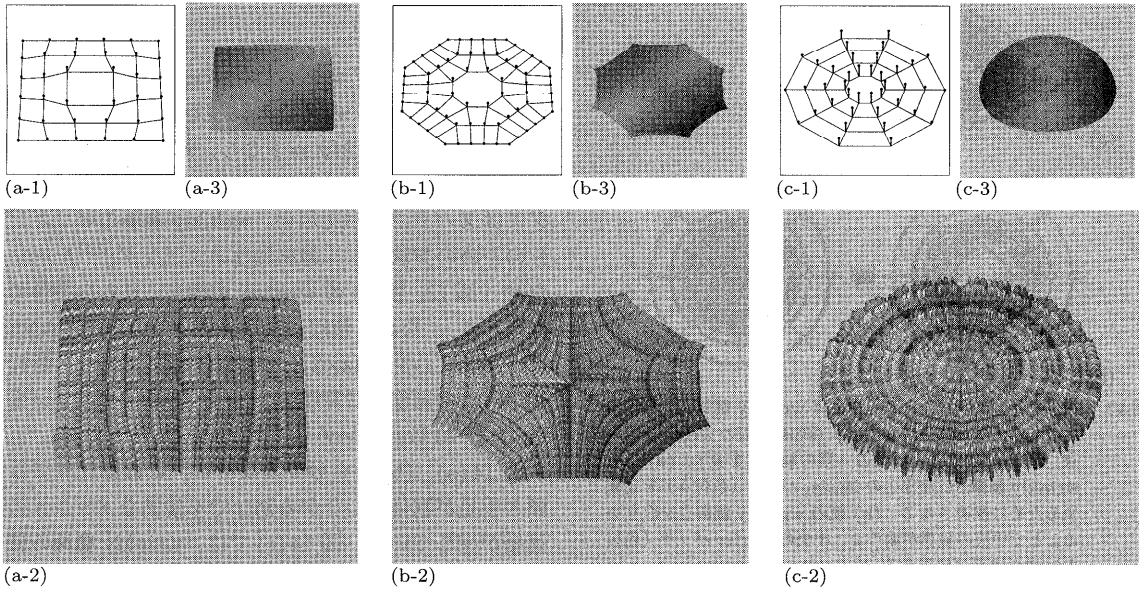
図 14 (a), (b) から分かるように, このモデルでは頭と手足が胴体につながる部分で不規則なメッシュの連結が生じている。図 14 (c) では, wrinkly 曲面の持つ「空間/スケール方向の局所的類似性」により, 大きな変位がより小さないくつかの変位から構成され, その階層構造が毛皮のスケール方向の特徴をうまく表している。そして, 必要以上にポリゴン数を増やすことなくその構造が続いていくように見える効果を生み出している。この例から分かるように, wrinkly 曲面は, 基本的に 2 次元面であるにもかかわらず, 3 次元的な効果***を生み出すことができる曲面であるといえる。

Wrinkly 曲面のように曲面に凹凸を与える CG 技法として, テクスチャマッピングやバンプマッピング等が考えられる。CG 技法としての観点からそれら従来技法と比較した wrinkly 曲面の主な利点として, 1) 操作性の良さ, 2) LOD (Level Of Detail) の実現, があげられる。1) は, wrinkly 曲面はあくまで制御点の補間により定義される曲面であり, 各 wrinkly 制御点に与えた高さ値や色値等を直観的に変えることで, 曲面の凹凸の具合や色の変化を (「空間/スケール方向の局所的類似性」を維持しながら) 局所的に変更可能であることを指す。また, 2) は, 「適応的サンプリング」により, 曲面モデルそのものを変更することなく, 視点位置に依存した適切な詳細度で画像が生成できると

*** 適応的サンプリングとは, a) スクリーン上に投影される大きさが適切なこと, b) 「空間/スケール方向の局所的類似性」を適切な形で維持すること, c) 視点の位置によって曲面の見え目が変わらないこと, という条件を満たすように wrinkly 曲面上の点を適切にサンプリングしてポリゴン群を生成する方法として, 著者らが文献 14) で提案したものである。

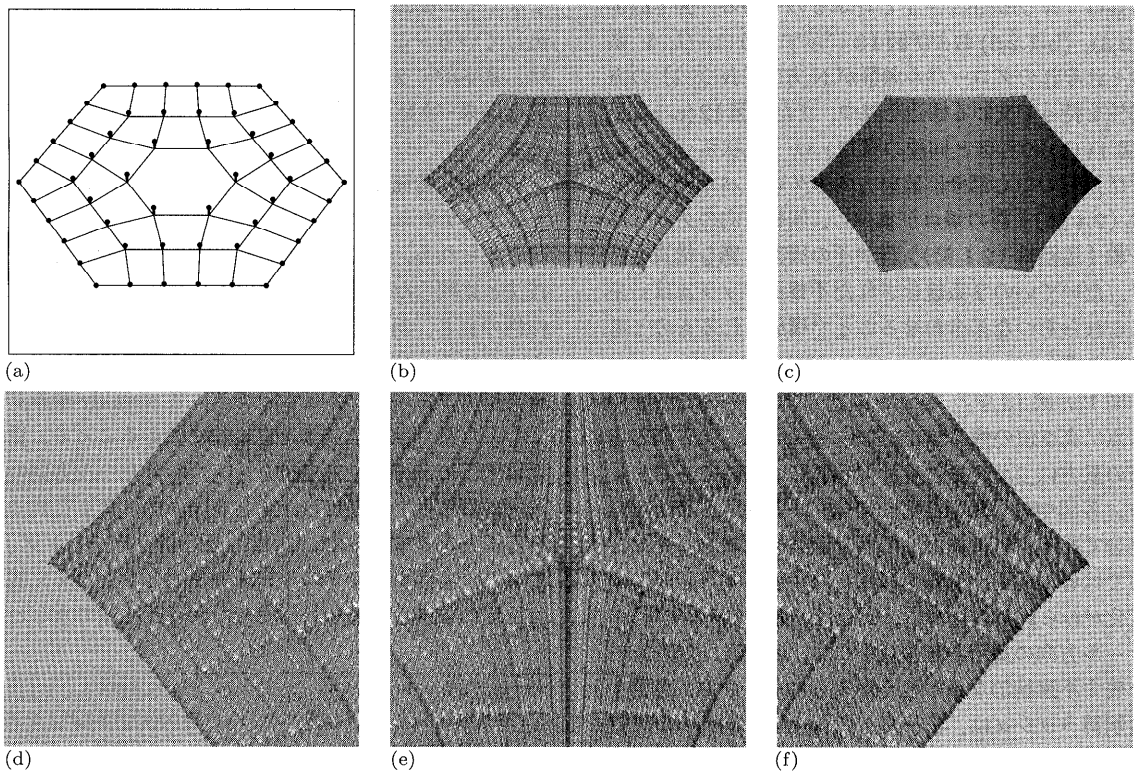
*** 図 14 のモデルでは, 各 wrinkly 制御点に高さ値の他に透過率を与えて IST を適用することで, 3 次元的な質感を生み出している。

* 実際には, これらだけではなく, たとえば図 10 (d') のメッシュは螺旋形の構造をしている。



制御メッシュ(a,b,c-1)に対する wrinkly 曲面 (a,b,c-2) と補間曲面 (a,b,c-3).
 (a-1,2,3) は 2 方向ネット, (b-1,2,3) は 4 方向ネット, (c-1,2,3) は環状ネット. 2 方向ネットは正方メッシュと同じ構造である.

図 12 2 系統化された不規則メッシュの構造
 Fig. 12 Structure of irregular mesh with two-grouped global axes.



制御メッシュ(a)に対する wrinkly 曲面 (b) と補間曲面 (c). (d), (e), (f) は (b) の左端部, 中央部, 右端部をそれぞれ 3 倍に拡大したもの. 「適応的サンプリング」(文献 14) 参照) により LOD (Level Of Detail) が実現されている. Wrinkly 曲面上の「空間/スケール方向の局所的類似性」の構造が確認できる.

図 13 3 方向ネットに対する wrinkly 曲面
 Fig. 13 Wrinkly surface on 3-directional net.

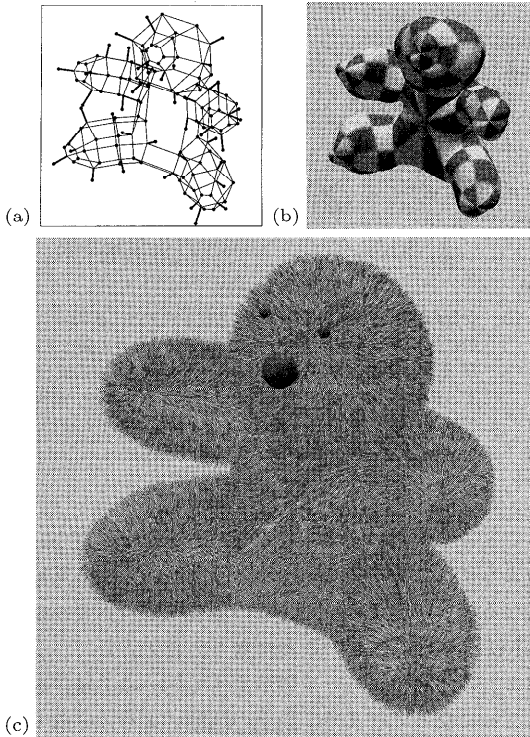


図 14 Wrinkly 曲面を用いた形状モデルの例: (a) 制御メッシュ (格子点上に wrinkly 制御点が位置し, 先の丸いトゲで高さ値分の法線方向への持ち上げを表す). (b) 各 wrinkly 制御点に対する 3 次 Bézier パッチ. 隣接するパッチを互いに色分けして示している. LST は適用せず高さ $H(t)$ 分をそのまま法線方向に持ち上げて表示した. (c) wrinkly 曲面 (色は 1 色とし, 鼻と目は別データとして付加).

Fig. 14 An example of shape model by using wrinkly surface.

いう利点である. 実際, 図 13(d), (e), (f) では, 拡大の下で, (b) の特徴を損なうことなく適切な LOD が実現されている*.

5. 結 論

本論文では, 補間曲面とフラクタルとを融合した wrinkly 曲面を不規則メッシュ上に構築する方法に関する提案を行った.

不規則メッシュ上での反復交差変換を実現するため, 不規則メッシュに対して「グローバル軸の 2 系統化」の制約条件を与え, 2 次元反復交差変換の一般化を行った. そして, それをもとに, 正方メッシュの場合を拡張した形で, 制約付き不規則メッシュ上で「空間/スケール方向の局所的類似性」の構造を持つ wrinkly 曲面を

形成した. その構造は, 制約付き不規則メッシュを代表する N 方向ネットや環状ネットに対する wrinkly 曲面上で実際に確認することができた. そして, その応用例として, 不規則なつながりを持つメッシュによる具体的な形状モデルの作成を行い, wrinkly 曲面が形状モデリングの一手法として有効なものであることが確認できた.

今後の方針として,

- 「グローバル軸の 2 系統化」の制約条件を取り除いた, 任意の不規則メッシュ上での 2 次元反復交差変換の実現.
- 4 角パッチ以外の任意の多角形パッチに基づく wrinkly 曲面の定義.
- 反復交差変換の反復ごとに階層化, 構造化した wrinkly 曲面の提案.
- 連続性のある (滑らかな) 曲面形状も実現するための一般化.

等を目指し, wrinkly 曲面のさらなる一般化を考えていく必要がある.

謝辞 本研究の一部は文部省科学研究費補助金 (基盤 C 一般 11680369) の援助を受けている.

参 考 文 献

- 1) Farin, G.E. *Curves and Surfaces for Computer Aided Geometric Design: A Practical Guide*, 2nd ed., Academic Press, Boston, Massachusetts (1990).
- 2) Watt, A. and Watt, M.: *Advanced Animation and Rendering Techniques: Theory and Practice*, ACM Press, New York (1992).
- 3) Ebert, D.S. (Ed.): *Texturing and Modeling: A Procedural Approach*, Academic Press, Boston, Massachusetts (1994).
- 4) Peitgen, H.O. and Saupe, D. (Ed.): *The Science of Fractal Images*. Springer-Verlag, New York (1988).
- 5) Barnsley, M.F.: *Fractals Everywhere*, 2nd ed., Academic Press, Boston, Massachusetts (1993).
- 6) Demko, S.: Construction of Fractal Objects with Iterated Function Systems, *Computer Graphics (SIGGRAPH '85)*, 19, No.3, pp.271-278 (1985).
- 7) Miller, G.S.P.: The Definition and Rendering of Terrain Maps, *Computer Graphics (SIGGRAPH '86)*, 20, No.4, pp.39-48 (1986).
- 8) Oppenheimer, P.E.: Real Time Design and Animation of Fractal Plants and Trees, *Computer Graphics (SIGGRAPH '86)*, 20, No.4, pp.55-64 (1986).

* 毛皮の 3 次元テクスチャを扱った文献 [12] では, 拡大時の質感の維持が困難であることが問題とされている.

- 9) Barnsley, M.F., Jacquin, A., Malassenet, F., Reuter, L., and Sloan, A.D.: Harnessing Chaos for Image Synthesis, *Computer Graphics (SIGGRAPH '88)*, 22, No.4, pp.131-140 (1988).
- 10) Musgrave, F.K., Kolb, C.E., and Mace, R.S.: The Synthesis and Rendering of Eroded Fractal Terrains, *Computer Graphics (SIGGRAPH '89)*, 23, No.3, pp.41-50 (1989).
- 11) Szeliski, R. and Terzopoulos, D.: From Splines to Fractals, *Computer Graphics (SIGGRAPH '89)*, 23, No.3, pp.51-60 (1989).
- 12) Kajiya, J.T.: Rendering Fur with Three Dimensional Textures, *Computer Graphics (SIGGRAPH '89)*, 23, No.3, pp.271-280 (1989).
- 13) 藤本, 大野: 形状と質感の融合のためのモデリング法, 情報処理学会グラフィクスとCAD 研究報告, No.84, pp.13-18 (1997).
- 14) 藤本, 大野: Wrinkly 曲面—手続きの補間によるパラメトリック曲面の形成, 情報処理学会論文誌, Vol.39, No.7, pp.2168-2179 (1998).
- 15) 藤本, 大野: 不規則メッシュ上での wrinkly 曲面の形成, 情報処理学会グラフィクスとCAD 研究報告, No.88, pp.13-18 (1997).

付録 「グローバル軸が 2 系統以下に系統分けされる」ことの証明

(証明) メッシュ上のローカル軸を連結することでグローバル軸がつくられる。ここで、メッシュ上の異なる 2 つのローカル軸 A, B について、 A から出発して、連結しているローカル軸上を渡り歩いて B まで到着することを考える (図 15(a))。その際、 A から B までの経路 L 上のローカル軸を系統分けしていく。まず、 A が U 系統に属すると決め、同じメッシュ格子点上で A と交差するローカル軸 A' が V 系統に属するとする。すると、 A から出発後、 A と同じグローバル軸上にあるローカル軸はすべて U 系統、それらに交差するローカル軸は A' と平行なのですべて V 系統となる (図 15(a) 中の (1) の部分)。経路 L 上で曲がり角に来た (すなわち、交差する別のグローバル軸に移る) とき、今度は渡り歩くべきローカル軸が U 系統から V 系統に変わり、それらに交差するローカル軸が U 系統となる (図 15(a) 中の (2) の部分)。そして、次の曲がり角では再び U 系統上を渡り歩くことになり、以下、同様に U 系統と V 系統が交互に繰り返される。そして、 B に到着したとき、 B は U 系統か V 系統のいずれかに属することが決定される。ここで、メッシュ上の任意のローカル軸を上記の B と考えることにより、メッシュ上の任意のローカル軸は U 系統か V 系統の 2 系統のいずれかに属す

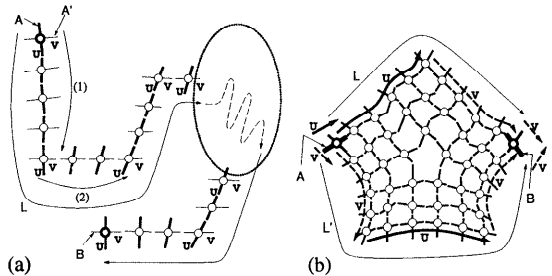


図 15 付録の説明図: (a) A 軸を U 系統とすると、 B 軸は V 系統に属することが決定される。図中のローカル軸は太線を U 系統、細線を V 系統とする。(b) A 軸を U 系統とすると、経路 L によれば B 軸は V 系統に属するが、経路 L' では B 軸は U 系統に属するように決定されてしまう。
Fig. 15 Figure for appendix.

ることになるといえる。しかし、たとえば、図 15(b) のように、 A からの経路の違いによって B の系統が異なる場合が起こりうる。この場合にはローカル軸を 2 系統に分離できないことになり、結果として、メッシュ上の任意のローカル軸がすべて同一の系統に属することになってしまう。以上の議論により、本論文で対象とする任意のメッシュに関して、各ローカル軸を連結したグローバル軸は、2 系統あるいは 1 系統 (すなわち、2 系統以下) に系統分けされることが分かる。

(平成 10 年 7 月 23 日受付)

(平成 11 年 9 月 2 日採録)



藤本 忠博 (正会員)

昭和 41 年生。平成 4 年慶應義塾大学大学院理工学研究科計算機科学専攻修士課程修了。同年 (株) 三菱総合研究所入社。平成 7 年慶應義塾大学大学院理工学研究科博士課程入学。平成 11 年単位取得退学。同年岩手大学工学部情報工学科助手。コンピュータグラフィクス、特に形状モデル理論に興味を持つ。



大野 義夫 (正会員)

昭和 45 年慶應義塾大学大学院工学研究科修了。同年同大学情報科学研究科助手。昭和 62 年助教授。平成 7 年より同大学理工学部計算機科学専攻教授。コンピュータグラフィクス、DTP 等に関連したアルゴリズムに興味を持つ。工学博士。