

4分木構造を用いたトリム曲面の分割手法

藤村 真生[†] 小堀 研一[†] 久津輪 敏郎[†]

本論文では、CGにおいて穴の開いた曲面やトリムされた曲面を表示する際に、曲面分割を最小限に抑えて高速に処理する手法を提案する。現在、平面を高速表示するハードウェアは多く開発されているが、曲面の表示の際には、一般に近似的な平面に分割するという前処理が必要となる。本論文では拡張した4分木構造を用い、曲面の曲率だけでなくトリムされた曲面の境界を平面分割と曲面表示の評価の対象とする。トリムされた境界部分は、曲面上にマッピングされた3次のBézier曲線を用いて定義する。本手法では、曲面の分割を行う際に面とBézier曲線との交差状態を分類し、曲率の計算とともに分割の条件に適用する。この際、簡潔な交差状態にある面についてはその状態を4分木のデータとして記録し、分割を行わないことにより、同品質の表現力を持つ細かい面に分割する場合と比較して高速に処理することが可能となる。表示の際には、分割の処理において4分木のデータに記録された交差状態を利用することによりトリムされた部分を多角形で表現する。

A Method for Dividing Trimmed Curved Surfaces by Using Quad-Tree Structure

MASAO FUJIMURA,[†] KEN-ICHI KOBORI[†] and TOSHIRO KUTSUWA[†]

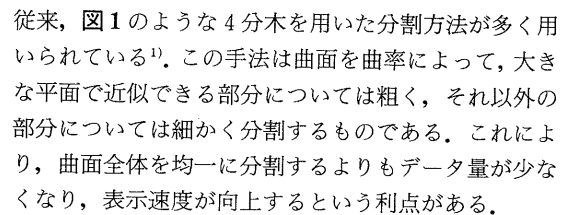
This paper describes an algorithm for the polygonal subdivision of parametric trimmed surfaces. Today, there are many graphics hardwares that can display polygons (patches) very fast. It is necessary to divide parametric surfaces into many fine patches in order to display surfaces using the graphics hardware. Former methods have problems displaying trimmed surfaces with complex boundaries or surfaces with tiny holes, because the parametric surface is divided into patches on the basis of only the curvature of surface. In the method presented here, the surface is divided into patches on the basis of both the curvature of surface and the complexity of the surface boundary. As a result, various size of patches are generated. Smaller patches are generated at only the complex boundaries of the parametric surfaces, and coarse patches are generated at the simple boundaries of the surfaces. We extend quad-tree data structure to represent boundaries of trimmed surface and holes. At the process of subdivision, the patch is classified according to the status of intersection between the patch and boundary curves and the method decides whether subdivision process is executed or not. As a result, the proposed method reduces the number of patches and the computational time.

1. ま え が き

近年ハードウェアの高速化に伴い、CGの分野においてはポリゴン表示のための専用ハードウェアが開発されている。これらは従来のようなグラフィックワークステーションに搭載されるのみならず一般家庭用のゲームマシンにも搭載され、表示は毎秒数万ポリゴンを上回る速度をも実現している。

ところがそれらは平面の表示を前提に開発されているため、曲面を表示する際には細かい平面に分割して近似的に表示する必要がある。

曲面の平面分割はその細かさに比例して表現力が向上するが、同時に計算量も比例して多くなる。そこで

従来、のような4分木を用いた分割方法が多く用いられている¹⁾。この手法は曲面を曲率によって、大きな平面で近似できる部分については粗く、それ以外の部分については細かく分割するものである。これにより、曲面全体を均一に分割するよりもデータ量が少なくなり、表示速度が向上するという利点がある。

この手法は、4分木データ構造を用いた場合の最も単純な方法であると考えられる。そこで本論文ではこれを従来法と呼ぶことにするとともに、本稿で提案する手法との比較対象とした。

また従来手法では、4分割する場所を任意の位置に設定することにより、効率的な分割を行うなどの改良もなされている²⁾。ところがこれらの手法において、平面分割の評価は曲面の曲率に関する部分にのみおかれていたため、穴の開いた曲面やトリムされた曲面に

[†] 大阪工業大学工学部電子工学科

Faculty of Engineering, Osaka Institute of Technology

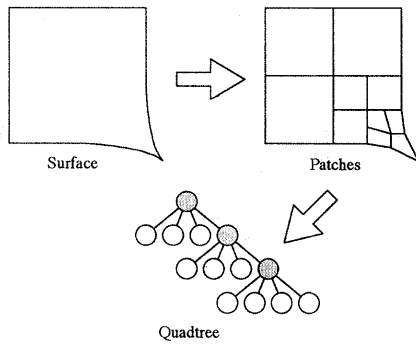


図1 4分木構造を用いた曲面分割
Fig. 1 A surface represented by quad-tree structure.

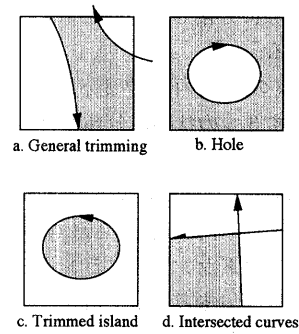


図2 曲線によるトリム部分の定義
Fig. 2 Definition of trimmed surface.

については考慮されていないのが現状である。

曲面を実際に表示する場合、その多くは対象となるモデルが現実存在する物体であると考えられる。現実存在する物体にはトリムされた部分や穴が多く存在する。したがって曲面上に開いた穴やトリム部分の表現はCGにおける重要な課題であると考えられる。

そこで本論文では4分木を用いた曲面の分割手法を拡張し、穴の開いた曲面やトリムされた曲面を表示する際に分割を最小限に抑える手法を提案し、その効果を検証する。

2. トリム曲面の定義

本研究では、曲面のトリムされた部分の表現を行うための処理を高速化することを目的としている。そこで、最初に曲面と曲面上のトリムされた境界とを定義する。

2.1 双3次 Bézier 曲面

定義される自由曲面にはCADにおいて広く用いられている双3次の Bézier 曲面を扱うこととした。

双3次 Bézier 曲面は次式により定義される。

$$p(u, v) = \sum_{i=0}^3 \sum_{j=0}^3 B_{i,3}(u) B_{j,3}(v) b_{ij} \quad (0 \leq u \leq 1, 0 \leq v \leq 1). \quad (1)$$

ここで $B_{i,n}(u)$, $B_{j,n}(v)$ は n 番目のベルンシュタイン基底関数である。また b_{ij} は曲面を定義する制御点群である。これらは x, y, z の3次元座標を持ち隣り合う点を結ぶことで 4×4 の曲面定義ネットを構成する。

2.2 トリム部分の定義

曲面上に開いた穴やトリムされた境界を定義するため、3次の Bézier 曲線を用意する。

3次の Bézier 曲線は、以下の式により定義される。

$$p(t) = \sum_{i=0}^3 B_{i,3}(t) b_i \quad (0 \leq t \leq 1). \quad (2)$$

ここで b_i は曲線を定義する2次元の u, v 座標系における制御点群であり、これにより曲線は u, v 座標を持つ2次元曲線として定義される。本手法ではこの曲線の2次元座標を上記双3次 Bézier 曲面のパラメータ u, v にそれぞれ対応させてマッピングする。このため曲線は各座標とも0から1の範囲で定義することとした。

本手法では曲線の始点から終点に向かう方向から見て左側を曲面の存在する部分、右側を曲面の存在しない、つまり穴やトリムされた部分として定義することとする。また定義されるトリム部分は次に示す規則に従うものとする。

- トリム部分を表わす曲線の始点および終点は、図2aに示すように曲面の外側か、あるいは境界上に存在しなければならない。

- 右向きに定義された閉じた曲線は、同図bに示すように曲面に開いた穴を表わす。

- 穴と逆向きに定義された閉じた曲線は、同図cに示すようにその外部をトリムされた部分として扱う。

また、本手法ではトリムされた部分の情報を優先的に扱うこととする。つまり、同図dに示すように曲線が複数存在する場合は、曲面を穴やトリム部分で順に切り取って最後に残された部分のみを面の存在する部分として扱う。

3. データ構造

本手法では、分割された平面データを管理するために、4分木構造³⁾を拡張して用いている。この拡張について、従来法と比較しその効果を検証する。

3.1 4分木構造

4分木構造をトリムされた曲面の平面分割に適用さ

せた場合、実際に分割して生成される平面は図3aに示すようになる。

4分木の各々の節(以下、クオドラントという)はそれぞれ次に示す3つの状態に分類される。

- A. 分割されている。
- B. 分割されず、平面が表示される。
- C. 分割されず、平面が表示されない。

Aのようなクオドラントをブランチと呼び、BおよびCのクオドラントをリーフと呼ぶ。ここで実際の表示に使われるクオドラントはBのみである。

また、Aのクオドラントが分割されて生じる4つのクオドラントとの関係を親子関係で表現し、任意のクオドラントに達するまでに行われた分割の数を、そのクオドラントのレベルと呼ぶ。ここで4分木構造の元となる、分割された平面全体に相当するクオドラントはレベル0である。

クオドラントはそれぞれ分割された平面に対して一意に対応する。任意のクオドラントに対応している平面を、以下パッチという。

3.2 拡張した4分木構造

本手法では、上記の3つの状態に次に示す状態を追加することにより、新しいデータ構造によるトリム部分の表現を実現した。

- D. 分割されず、トリム部分の境界を表わす。

この状態を導入することによって得られる実際の分割を図3bに示す。このクオドラントに対応するパッチの内部には、面の存在する部分と存在しない部分とを区切るための境界線のデータを付加する。

この状態を導入することにより、パッチと曲線との交差が単純である場合はパッチを分割することなく境界部分を表現することができる。

図3から従来法と比較して次の点で優れていることが確認できる。

- ・分割数を抑え、面数が少ない。
- ・4角形で分割しているために生じるエイリアスを生じない。

ところで、分割時の計算量と表示の際の品質の良さ

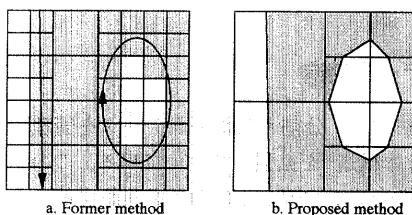


図3 トリム部分の分割パッチによる表現
Fig. 3 Trimmed area represented by divided patches.

とは相反関係にある。従来法では、画像の質を問題にしなければ処理時間を大幅に減少させることができ、逆に処理時間を問題にしなければ本手法で扱う前記のDの状態を用いた分割を行うよりも質の良い画像を生成することも可能である。

そこで本研究では、最終的に生成された画像を目視により比較してほぼ同レベルの質であると判断できるものについて、その処理時間と面数により従来法と本手法とを比較することにした。

4. 曲面の分割と表示

一般に木構造のデータを構築する場合、最初に基本となるデータを生成し、再帰的な処理により行う。4分木構造を用いた曲面の平面分割法では、最初に曲面全体を1つのパッチと考慮してクオドラントを生成する。そして図4に示すように、対象となるクオドラントに対して判定・分割の手順を再帰的に行うことにより分割の処理を行う。

この手順は、分割の必要があるかどうかについての判定処理と、その結果に応じて行われる再帰処理とに分けられる。判定処理では、クオドラントのレベルとパッチの曲率および曲線との交差状態での判定を順に行い、再帰処理では子の生成と分割手順の呼び出しおよび親のループに戻る処理を行う。

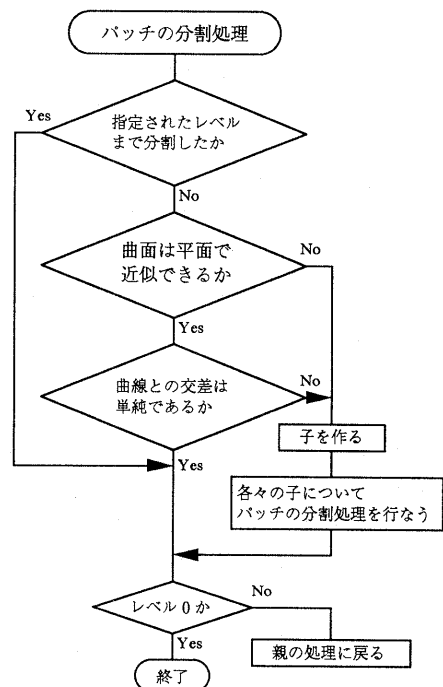


図4 分割処理の流れ
Fig. 4 Flow chart of patch division.

表示の際にも、分割の処理で作られた木を同じように再帰的に検索する。そしてリーフにたどり着いたところでパッチを描画する処理を繰り返す。

4.1 分割の判定

本手法における曲面の分割の判定は、対象となるパッチの次の3点に着目して行う。

4.1.1 分割の深さ

1点目は、分割のレベルがあらかじめ設定された最高レベルまで達しているかどうかである。最高レベルは求められる画像の品質と曲面の形状により経験的に設定する必要がある。

4.1.2 曲率の計算

2点目は、曲面の曲率から対象となるパッチを平面とみなすことが可能かどうかという点である。

この判定は図5に示すパッチの4頂点での法線ベクトル N_i , u 方向の接線ベクトル T_{iu} , v 方向の接線ベクトル T_{iv} とパッチ中心での法線ベクトル N_c の各々から次に示す4式によって行う。

$$1 - T_{iu} \cdot T_{ju} < e (i, j \in \{0, 1, 2, 3\}, |i - j| = \text{odd}). \tag{3}$$

$$1 - T_{iv} \cdot T_{jv} < e (i, j \in \{0, 1, 2, 3\}, |i - j| = \text{odd}). \tag{4}$$

$$1 - N_i \cdot N_j < f (i, j \in \{0, 1, 2, 3\}, |i - j| = \text{odd}). \tag{5}$$

$$1 - N_i \cdot N_c < f (i \in \{0, 1, 2, 3\}). \tag{6}$$

ここで e, f はそれぞれ edge linearity, flatness と呼ばれる定数である⁴⁾。また T_{iu}, T_{iv}, N_i, N_c は、それぞれ図5に示す単位ベクトルである。

式(3)および式(4)ではベクトルの内積の計算を合

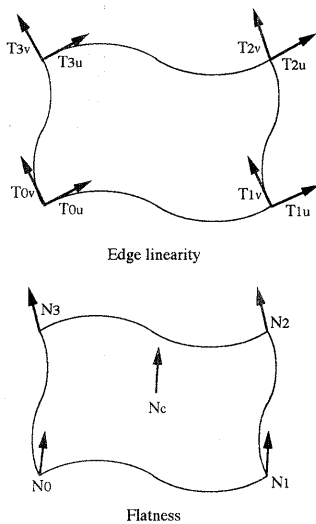


図5 曲率による分割のためのベクトル
Fig.5 Vectors of division by curvature.

計で最大8回行うことにより、パッチの境界部分の直線性を判定する。また式(5)および式(6)ではベクトルの内積の計算を合計で最大10回行うことにより、パッチの平坦性を判定する。これらの式がすべて真となった場合、パッチは充分平面に近いものと認め、分割の必要なしと判断する。

なお、この判定において使用される定数 e, f のそれぞれの値は、曲面の形状や必要な画像の品質、近似精度に応じて定める必要がある。本研究で行った実験では $e=f=0.15$ とした。

4.1.3 曲線とパッチとの干渉判定

3点目は、曲線によって定義されたトリム部分と穴を表現するのに対象パッチを分割する必要があるかどうかという点である。この判定は、対象とするパッチについて図6に示す手順で各々の曲線を1本ずつ順番に取り上げて行う。この処理は、まず対象パッチと曲線との干渉判定を行う。

そして曲線との干渉があるパッチに対しては曲線とパッチの境界との交差状態を算出し、次項に示す判定を行って3章で述べたA, B, C, Dの4状態に分類する。

曲線との干渉がない場合は、パッチの内部に曲線を完全に含んでいるかどうかの判定を行う。これは曲線の始点がパッチの内部にあるか否かで判定できる。

パッチが曲線を完全に含んでいる場合この曲線で定義された部分を表現するためには、パッチを平面分割しなければならない。したがってこのパッチはAの状態に分類される。

内部に曲線がない場合は、パッチが曲線の左右どちら側にあるかの判定を行う。

パッチが曲線の左側にある場合、Bの状態に分類さ

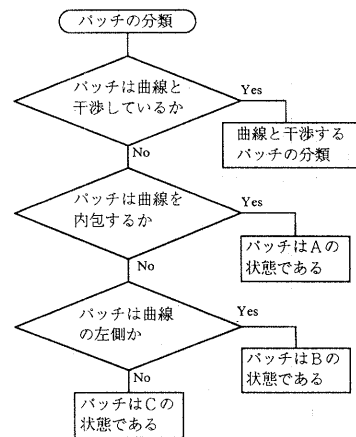


図6 パッチの分類処理の流れ
Fig.6 Flow chart of classification of patch.

れる。このパッチは曲線との干渉がないので平面分割する必要もなくそのまま全体を描画すればよい。同様にパッチが曲線の右側にある場合、Cの状態に分類される。このパッチはBに分類されたパッチの逆で完全にトリムされた部分である。したがって分割する必要はなく、また描画する必要もない。

4.1.4 曲線と干渉するパッチの分類

パッチが曲線と干渉している場合、曲線とパッチの境界との交差状態によってパッチの詳細な分類を次のように行う。なお、以後の判定においてDと判断されたパッチは、従来法においては表現不可能であるため便宜上Aに分類する。

i) パッチを横断する曲線が1本の場合

・図7aに示すように曲線がパッチを1度だけ横断する場合、このパッチはDの状態である。

・曲線が図7bに示すように2回以上パッチを横断する場合、このパッチはAの状態である。

・図7c, dでは同じ曲線が対象とするパッチと干渉している。同図cではトリム部分をパッチのDの状態に分類することが可能であるが、同図dでは曲線がパッチの同一の境界から出入りしているため、Dの状態では表現することができない。したがってこのような場合はAの状態に分類し、さらに分割を行う。

ii) パッチを横断する曲線が複数の場合

・曲線が相互に交差せず、図8aに示すように一方の曲線によって切り残される部分が他方の曲線によって完全に切り取られてしまうような場合、このパッチはCの状態である。なぜなら、2.2節で示したトリム部分を優先的に扱うという規則に基づいている。

・ある曲線が別の曲線の右側にあり、2つの曲線が

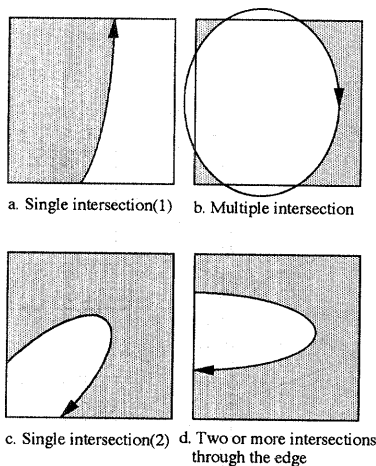


図7 交差する曲線が1本の場合
Fig. 7 In case of one curve across the patch.

同じ方向を向いている場合、図8bに示すように右側の曲線を無視し、左側の曲線とパッチの境界との交差を計算してこのパッチをDの状態に分類する。

・曲線が相互に交差せず、図8cに示すように一方の曲線により切り残された部分を他方の曲線が切断するような場合、これらの曲線によって切り残された部分をDの状態を用いて表現することはできない。したがってこのパッチはAの状態に分類する。

・曲線がパッチの中で図8dに示すように相互に交差する場合、このパッチはAの状態に分類する。

4.2 データの組み込み

分割の判定を終了し、リーフとして決定されたクオドラントは、その状態をデータとして保存することによりトリムされた部分を表現するために用いる。先の判定でAの状態に分類されたパッチの中で最高レベルになったものはBの状態として扱う。

ただし、曲線がパッチの内部で交差している場合は例外とする。図9において太枠で囲んだ部分がこの場合に当たる。この部分をBの状態として描画すると、

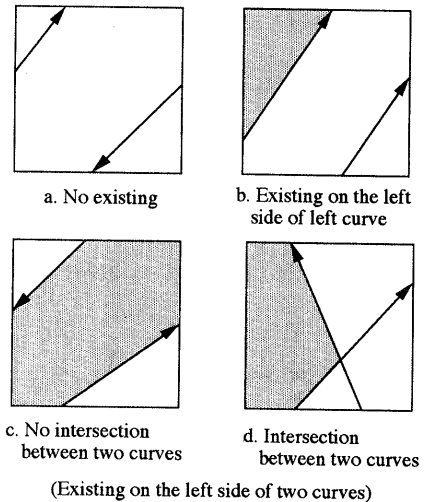


図8 交差する曲線が複数の場合
Fig. 8 In case of plural curves across the patch.

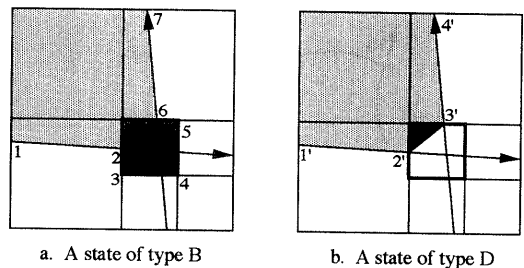


図9 TypeBをTypeDに変換
Fig. 9 Change type B into type D.

隣り合うパッチの境界線との整合が取れず、同図 a 中の頂点 1 から 7 で示したような形状として描画されてしまう。そこで同図 b に示すように複数の曲線によって切り残された部分を算出し、D の状態のパッチとして扱うことによって頂点 1' から 4' で示すような形状として描画することとした。

データの組み込みに際し、B の状態のクオドラントと C の状態のクオドラントとに与えられるデータは単純であり、そのクオドラントを表示するか否かの情報だけを持たせればよい。

D の状態のクオドラントはトリムされた部分や穴を表現するためのクオドラントであるので、曲線とパッチの辺との交点の座標を算出し、曲線の方向とともにクオドラントのデータとして記録する。

4.3 表示

表示の際、実際の描画に用いるのは B の状態のクオドラントと D の状態のクオドラントである。

B の状態のクオドラントは 4 角形である。本来ならば捻れを考慮して、4 角形を 3 角パッチに分割して表示しなければならない。

しかし本研究では、以下の点から分割を行っていな

い。

- 分割の際にすでに曲率による分割をしているため実用的には平面に近似できると考えられる。

- 3 角形に分割して表示するとパッチの数が 2 倍になり、描画速度の低下の原因となる。

D の状態のクオドラントは曲線とパッチの各辺との交点の座標を持っている。これらは曲線の方向をもとにあらかじめ曲線がパッチから出ていく点と入ってくる点の識別情報を持たせる。

曲線がパッチから出ていく点を最初の点とし、曲線の左側にあるパッチの頂点を左回りに結んでゆく。そして曲線がパッチに入る点を最後に結ぶと、曲線の右側は描画されず、左側だけが描画されることになる。

またこの場合にも、B のパッチと同様の捻れが発生する可能性がある。しかし、前述の理由により同様に 3 角形に分割せずに描画する。

5. 実験結果

本論文で提案する平面分割および表示の高速化を検証するため、次に示す 4 種類の形状データにより実際に分割および表示を行い、従来法と比較した。

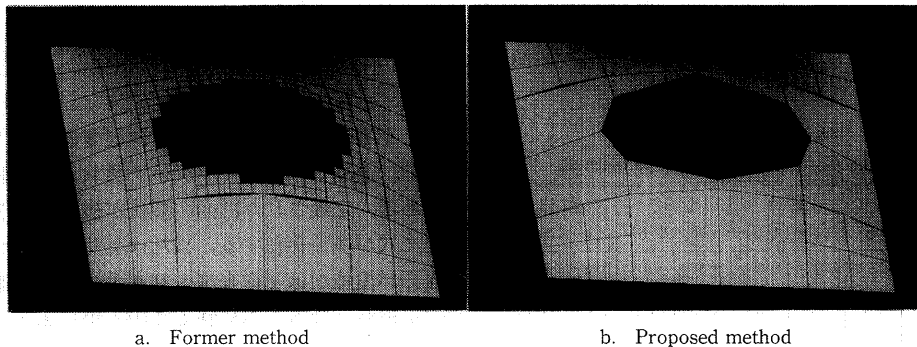


図 10 曲面上に単一の穴
Fig. 10 A hole on a curved surface.

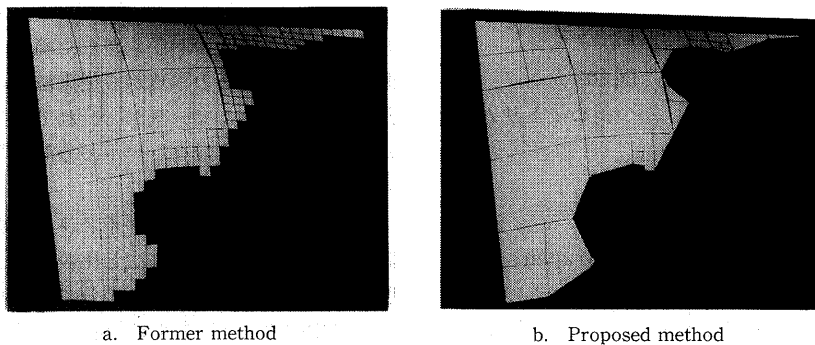
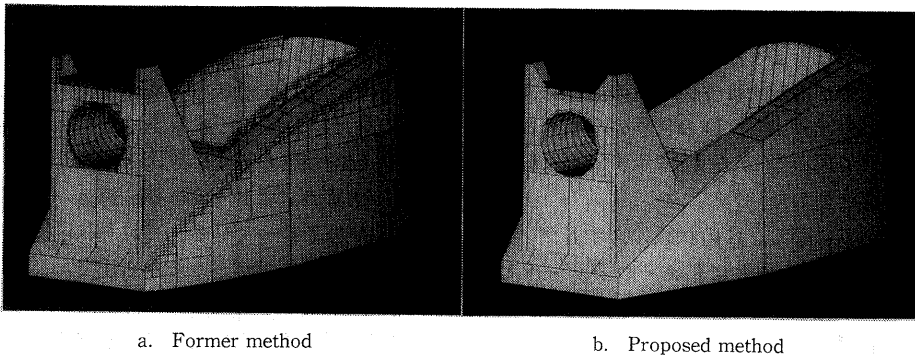


図 11 曲面に 2 つの穴とトリム境界
Fig. 11 Two holes and a trimming boundary curve on a curved surface.

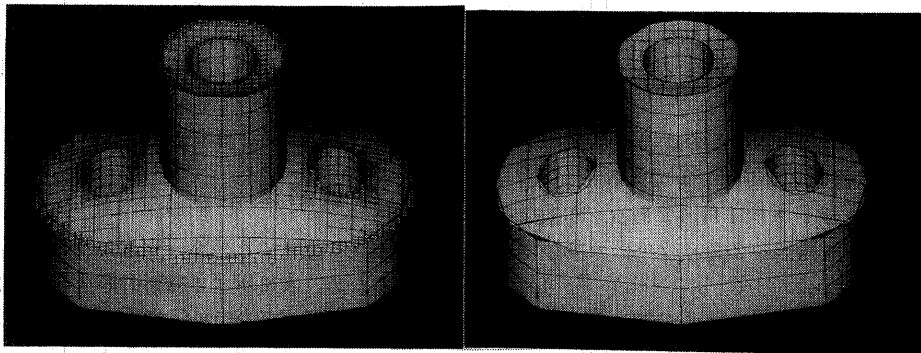


a. Former method

b. Proposed method

図12 テープカッター形状

Fig. 12 A model of a tape cutter.



a. Former method

b. Proposed method

図13 フランジ形状

Fig. 13 A model of a flange.

表1 従来法と本手法との比較

Table 1 Statistics for execution of the method.

| | 図10 (レベル5) | | 図11 (レベル5) | | 図12 (レベル6) | | 図13 (レベル6) | |
|-------------|---------------|------|---------------|------|---------------|------|---------------|------|
| | 従来法 | 本手法 | 従来法 | 本手法 | 従来法 | 本手法 | 従来法 | 本手法 |
| パッチ数 | 148 | 28 | 572 | 58 | 2166 | 569 | 2328 | 652 |
| クオドラ ント数 | 261 | 37 | 1157 | 145 | 4156 | 608 | 4075 | 875 |
| 分割処理 時間 | 1.00 | 0.20 | 1.00 | 0.10 | 1.00 | 0.12 | 1.00 | 0.06 |

(処理時間は、従来法を1とする.)

- ・単一の曲面に穴を開けた形状 (図10)
- ・単一の曲面に穴を2つ開け、かつ斜めにトリムした形状 (図11)
- ・セロテープカッター形状 (図12)
- ・フランジ形状 (図13)

分割および表示の実験は、Silicon Graphics社のIndyを用いて行った。なお図10~13はパッチのワイヤフレーム図と陰影図とを重ね合わせて表示している。

実験結果を表1に示す。表1は従来法と提案した手

法とを適用し表示したときに同程度の品質である画像での、最高分割レベル・データ数および処理時間の比である。定義された曲面の形状が複雑、すなわち曲面上に穴が多く存在する、またはトリムされた曲面の境界部が複雑なほど従来法より本手法の処理速度の高速化が顕著となる。本実験では表1から1/5~1/15の高速化が得られている。

図10,11の写真から従来法では穴の部分においてはレベル5の分割が全体に行われるのに対して本手法では一部だけにレベル5の分割がおよんでいることがわかる。また、曲面のトリムされた境界付近での分割状態をみると、図12の写真から従来法では実験で設定した最高レベル6まで分割が行われているが、本手法ではその一部だけに最高レベルの分割が行われていることがわかる。

表1からも生成されたクオドラント数および実際に表示するパッチ数は従来法と比較して1/6から1/5程度となっており、必要以上の分割を行わない少ないデータ量で従来法と同程度の品質の画像が生成できていることを示している。

6. む す び

実験結果より、4分木データ構造に本稿で提案した拡張を加えることによって、高速な処理および少ないデータ量でトリムされた曲面の表現ができることが明らかになった。

本手法では、分割する必要がある場合にはパッチの中心で4分割しているのので、トリム部分が曲面の一部に限られる場合などは、分割の影響によって広い範囲にわたり細分割が行われてしまう可能性がある。それを回避する方法として、分割する場所をパッチの中心だけでなく効果的に分割できる場所に設定することが考えられる。

また図10に見られるように異なるレベルの細分割を行っているため隣り合うパッチが同じレベルでないと、その間に多角形の隙間を生じてしまうことがある⁵⁾。そこで本論文で提案した手法を実際に用いる際には、この隙間を補間するための手法を用意する必要がある。

さらに、現在は最大分割レベルを形状に応じて経験的に設定しているが、今後はこれを形状の特性に応じて自動で設定できるように改良を加えていきたい。

参 考 文 献

- 1) Barnhill, R. E. and Kersey, S. N.: A Marching Method for Parametric SSI, *Computer Aided Geometric Design*, Vol. 7, No. 1-4, pp. 257-280 (1990).
- 2) 原田耕一, 森山真光: 最適平面近似のための双3次Bézierパッチの分割法, 電子情報通信学会論文誌 (D-II), Vol. J76-D-II, No. 3, pp. 1684-1690 (1993).
- 3) Hanan, S.: The Quadtree and Related Hierarchical Data Structure, *Comput. Surv.*, Vol. 16, No. 2, pp. 187-260 (1984).
- 4) Kosters, M.: Curvature-dependent Parameterization of Curves and Surfaces, *Computer Aided Design*, Vol. 23, No. 8, pp. 569-578 (1991).

- 5) Herzen, B. V. and Barr, A. H.: Accurate Triangulations of Deformed, Intersecting Surfaces, *Comput. Graph.*, Vol. 21, No. 4, pp. 103-110 (1987).

(平成6年11月4日受付)

(平成7年6月12日採録)



藤村 真生 (正会員)

1972年生. 1994年大阪工業大学電子工学科卒業. 現在同大学大学院博士前期課程において3次元モデリング, パーチャルリアリティーの応用に関する研究に従事. 電子情報通信

学会会員.



小堀 研一 (正会員)

1951年生. 1975年山梨大学大学院修士課程修了. 工学博士 (大阪府立大学). 1975年シャープ(株)に入社. 以後, 一貫してCAD, CAM, CGに関する研究開発に従事. 1991年大阪工業大学に奉職. 現在同大学工学部電子工学科教授. CAD, CG, パーチャルリアリティーの研究に従事. 著書「CAD・CG基本用語集」(工業調査会), 「三次元CG」(オーム社)など. システム制御情報学会, 精密工学会, 日本設計工学会, ACM各会員.



久津輪敏郎 (正会員)

1941年生. 1966年大阪工業大学電子工学科卒業. 1973年大阪府立大学大学院電子工学専攻博士課程修了. 工学博士. 同年大阪工業大学電子工学科講師, 助教授を経て, 1984年教授. 論理回路の自動合成, 電気電子用CADシステムなどの研究に従事. 「論理回路工学」(共著)など. 電子情報通信学会, プリント回路学会, IEEE各会員.