

輪郭線による曲面の隠面処理

小出 昭夫

日本アイ・ビー・エム株式会社 東京基礎研究所

面の境界をたどることによる隠面処理は、多角形データについて発展してきた。ここでは、この方法を、巻数と重複度の関係を利用したもの捉えることにより、曲面の場合に拡張する。視線と曲面が接する点のなす曲線、輪郭線に、その付随する曲面に沿って反時計回りに向きを与える。このとき、輪郭線を積分路として得られる巻数の和は、射影された曲面の重複度と一致する。したがって、隠面処理を、積分路の変更によって巻数の和を1にする操作とみなせる。このとき、曲面が多角形データの場合と異なるのは、次の3点である。(1) 視線の変更で輪郭線が動的に変わる。(2) 隠面を作る輪郭線が自分自身と交差することがある。(3) 隠面を作る輪郭線が巻込み点をもつことがある。上記の(2)と(3)のため、新しい規則を加える。また、巻込み点が生じる条件、分類、発見方法を与える。

Hidden Surface Removal for Curved Surfaces using Silhouette Traversal Logic

Akio KOIDE

IBM Research, Tokyo Research Laboratory
5-19 Sanbancho, Chiyoda-ku, Tokyo, 213 Japan

The boundary traversal method has been used to remove hidden surfaces of polygon data. We extended this method to handle curved surfaces.

In case of curved surfaces, we replace the boundaries by the silhouettes of the surfaces which are computed as the points tangent to the view direction. We set the orientation of the silhouettes so that they wind their associated trimmed surfaces counter-clockwise. We added new traversal rules because the silhouettes may intersect with themselves and because the silhouettes may have folding points of the surfaces. We classified the folding points into two types and gave the method to find them.

1. はじめに

この報告書の目的は、輪郭線による隠面処理の方法を曲面をもった図形に適用することである。隠面処理のアルゴリズムは、大別するとObject Spaceでの処理とImage Spaceでの処理とに分けられる^(1, 2)。ここで取り上げる方法はObject Spaceでの処理に属する。

ラスター式表示装置の普及した今日、その簡単さからImage Spaceでの隠面処理が主流となったかのように見える。しかしながら、CADのような応用分野では、(1) 1024x1024をはるかにこえる解像度や、(2) 写実的タッチより構成的象徴的表示や、(3) 可視(visible)物体だけをカソールで選択できることなどが要求されることから、輪郭線(silhouette)で可視曲面だけを切り取るObject Spaceでの隠面処理^(3, 4, 5, 6)が適しており、常によりよいアルゴリズムが求められている。

Object Spaceでの隠面処理は複合的技術であり、多面体データについて発展してきた。射影された多角形からなる輪郭線をたどる(traverse)ことによる隠面の切取は、WeilerとAthertonの論文⁽³⁾に始まる。しかし、その萌芽は、3次元上の有向線分を平面に射影し、射影面での重複度(contour)と線分の回転の向き(sense of rotation)を論じたAppel⁽⁷⁾に見出される。コンピュータによる実行には、これに加え、輪郭線の内点、外点の高速な判別方法、輪郭線同志の交点の能率的计算方法、適切なソート戦略が求められる。

ここでは、輪郭線による隠面処理を、曲面をもった図形に拡張するため、多様体の幾何学の視点から一般的な曲面について何が言えるか、まず、第2章で考察する。ここで多角形からなる図形と異なった事象がいくつか見だされるだろう。これにもつぎ、第3章では、簡単なプリミティブからなる図形に対し、具体的な適用方法を論じる。

2. 基礎理論

2. 1 曲線と曲面の幾何学

2. 1. 1 曲線/曲面の定義

n 次元位相多様体(topological manifold)は、その各点が n 次元Euclid空間の開集合に同相な近傍をもつHausdorff空間で一般に定義される⁽⁸⁾。

我々の隠面処理の目的には、もう少し限定された定義で十分である。 m 次元Euclid空間の点からなる集合の各点が、 n 次元Euclid空間の有界な開集合からの1対1連続な写像の像となる有界な近傍をもつとき、その集合を m 次元Euclid空間に埋めこまれた n 次元位相多様体と呼ぼう。連続性は、写像の各成分での連続性に置き換えられる。同様に、微分可能な多様体を微分が連続な1対1写像で定義する。

さて、埋めこまれた1次元多様体は曲線となり、埋めこまれた2次元多様体は曲面となる。しかし、Jordanの考えた曲線、区間 $[0, 1]$ から m 次元Euclid空間($m > 1$)への連続写像の像は、区間の両端以外で像が重複するなら、1次元多様体ではなくなる。

したがって、我々が実際にコンピュータ・グラフィックス

で扱う図形は、多様体の集まり、複体(complex)である。図形が1次元多様体と0次元多様体(点)からなり、各点がいずれかの1次元多様体の境界をなすとき、曲線と呼ぶことにする。前述の曲線で区間の両端で像が重複するなら、閉曲線と呼び、また両端以外で像が重複しないなら、単純(simple)と呼ぶことにする。

同様に2次元多様体と1次元多様体からなる図形が、各1次元多様体を2次元多様体の境界とすると、曲面と呼ぶ。

病的な曲線や曲面を、考察から取り除くため、図形は有限個の多様体からできているとする。

定義より、曲線、曲面の各部分は1次元の区間、2次元の領域からの写像の像で与えることができる。このような与え方をパラメトリック表示と呼び、もとの空間をパラメータ空間と呼ぼう。曲線や曲面の別の重要な与え方は、陰関数によるものである。すなわち、連続な関数 f にたいし、

$$f(x, y) = 0$$

$$\text{または } f(x, y, z) = 0 \quad (1)$$

を満たす点の集合で図形を定義する。

2. 1. 2 曲線/曲面への演算

隠面処理は射影を媒介にした曲線や曲面への演算として考えられる。射影された各点にたいし、視点から見一番手前の物を可視とする演算である。実際のグラフィックス生成では、さらに、可視とされた各点が、もとの曲線、曲面の属性(色やテクスチャなど)をひきついでいるとして画面に描かれる。

図形データは、埋めこみ先の空間での写像変換や、図形を定義するパラメータ空間での写像変換で、変形できる。このとき、写像変換が埋めこみ先の空間や定義するパラメータ空間の図形を含む領域で位相写像(1対1両連続写像)ならば、位相的性質(曲面の交差など)は変わらない。このような変形は、回転、移動、透視変換など、実際の問題にかなりあらわれている。

透視投影(perspective projection)は、図形データに透視変換を行ったあとで、射影を行うものと考えられる。図形データの中に視点が含まれない限り、図形の位相的性質は透視変換によって変わらない。

その他の曲線や曲面の演算として、図形データの合併、積、差などの集合演算がある。分子設計などの応用分野では、薬の活性部分の同定に集合演算が重要になる。

この章では、隠面処理の前に回転や透視変換が図形データに行なわれているものとして、今後、 z 軸を視線方向、 x 軸、 y 軸をそれに垂直な方向とする。また、上側(z 軸正方向)というとき、視点の側(視点への向き)をさす。

2. 1. 3 向きをもった曲線

2.1.1節では曲線が点集合として考察されていた。ここでは、向きをもった曲線を、曲線と写像の組に同値類をもうけたものとして定義し、その演算を導入する。

いま、2つの曲線が、それぞれ、区間 $[0, 1]$ から連続写

像 $C(t)$ と $C'(t)$ で与えられているとしよう。もし、この区間で単調連続増加関数 f をもちいて、

$$C'(t) = C(f(t)) \quad (2)$$

とできるなら、2つの曲線は同値とする。さもなくば、同じ像を与えても異なる曲線と考える。この定義より向きをもった曲線に一連の演算が定義できる。

曲線 C にたいし、逆向きの曲線 C^{-1} を $C(1-t)$ で定義する。曲線 C の制限 C' を

$$C'(t) = C(t_1 + (t_2 - t_1)t) \quad (3)$$

で定義する。ここで $0 \leq t_1 < t_2 \leq 1$ とする。曲線 C の終点と曲線 C' の始点が一致するとき、接続された曲線 C'' を

$$\begin{aligned} C''(t) &= C(2t) & t \leq 1/2 \\ &= C'(2t-1) & t > 1/2 \end{aligned} \quad (4)$$

で定義する。

2.2 輪郭線の発生と向き

2.2.1 発生機構

図形データからの曲線発生機構として、射影によって生じるものとそうでないものとに分類できる。折り目(稜)、曲面の交差などは、射影の以前にあったものである。ここで折り目は、曲面の微分不連続点の集まりである。

Appel⁽⁷⁾は、隠面処理を考える際に、射影面での図形データの重なりに注目し、図形の各面(face)が多角形からなるとき、その重複度が多角形の辺を境にして変わることを明らかにした。一般の曲面については、すでにある図形データの境界に加え、新たな輪郭線の発生という概念が必要になる。

まず微分可能2次元多様体を考えよう。この面で定義される法線ベクトル $\underline{n}$ と視線ベクトル $\underline{e}$ との内積 $\underline{n} \cdot \underline{e}$ の符号の変わり目(すなわち零点)は、新しい曲線(または曲面)を3次元図形データに付け加える。平均値の定理⁽⁹⁾より、射影面での重複度が変化するなら、この曲線の射影を境に生じることが導かれる。証明の概略は次で与えられる。

[証明] 微分可能2次元多様体の射影の重複度がその境界以外で変わる点の近傍で、視線方向(z 軸)に平行な切断面を考える。近傍が充分小さければ、図1の示唆するように、曲面と切断面との交点のなす曲線は、視線方向を定義域とする連続な関数 $f(z)$ とみなせる。重複度の多い側では、 $f(z_1) = f(z_2)$ となる2点がある。このとき、平均値の定理より、2点の間に $f'(z)$ となる点がある。この点では、内積 $\underline{n} \cdot \underline{e}$ が零となる。証明終り。

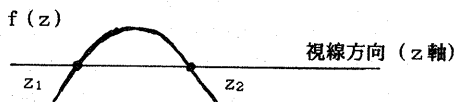


図1 重複度の変わる点の切断面

法線ベクトルと視線ベクトルの内積が零になる点は、パラメトリック表示では、パラメータ空間を (s, t) とすると、

$$\begin{vmatrix} \partial x / \partial s & \partial y / \partial s \\ \partial x / \partial t & \partial y / \partial t \end{vmatrix} = 0 \quad (5)$$

を満たす点集合で与えられる。陰関数表示では、

$$\partial f / \partial z = 0 \quad (6)$$

を満たす点集合で与えられる。

内積 $\underline{n} \cdot \underline{e}$ が零となる点の集合が面をなすこともある。この面の内点は、視線から見えず、2.3節で述べる隠面処理に関与しない。したがって、この場合、輪郭線に対応するものと3次元空間での曲線を、その面で境界で定義する。

向き付け不可能な面や折り目を含んだ面についても議論するため、微分可能2次元多様体の法線はその適当な近傍では連続に法線をとる、また、折り目をはさむときはその両側の法線にたいし折り目が逆向きとなるようにする。このとき、同様に、法線ベクトルと視線ベクトルとの内積の符号の変わり目で、射影面での重複度の変わる境界(輪郭線)に対応する曲線がもとの3次元空間に定義できる。

複体としての曲面が、3次元空間を内と外とに分ける場合、法線の向きを外にとることにより、内積 $\underline{n} \cdot \underline{e}$ が負の面を隠面として処理から除ける。

2.2.2 輪郭線の向き

前節で輪郭線が、射影面での重複度の変わる境界として導入された。したがって、輪郭線は、曲面の境界か、内積の符号の変化点の射影である。ここでは巻数の立場から輪郭線に向きを定義する。

2次元空間(射影面)での向きをもった曲線 C にたいし、与えられた点 P の周りの巻数(winding number)を積分

$$W(P) = (1/2\pi) \int_C d\theta(P) \quad (7)$$

で定義できる。ここで $\theta(P)$ は、曲線上の点が点 P にたいしなす角である。

巻数の重要性は、閉曲線にたいしては、その補集合の各連結成分で巻数の値が一定になることにある⁽⁹⁾。巻数は、点 P が閉曲線上のとき半整数値を取りえるが、それ以外では、整数値である。単純閉曲線の場合、その曲線が時計回りなら、内点の巻数は-1となり、反時計回りなら、+1となる。図2に閉曲線と巻数の例を示す。

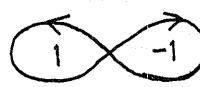


図2 a. 巻数の例-1



図2 b. 巻数の例-2

コンピュータによる実際の巻数計算では、角度を計算せずに、積分路を部分的に次々と簡単化することによって行なうか、基準半直線との交差を数えるのが速い。巻数を多角形データに適用すれば、付録1に示すように、Surany⁽¹¹⁾の閉曲線の内点/外点高速判別法が導ける。

輪郭線の向きが適切にとられれば、すべての輪郭線に対する巻数の和で、射影面の各点の重複度が求まると期待される。

このためには、輪郭線の向きはその付随するする面にたいし反時計回りとするればよい。ここで、輪郭線に付随する面が複数あるときは、2.3.1節で注意したように、それぞれの面に対して向きづけられた輪郭線があると考え。

この輪郭線の定義は、Sutherlandら⁽¹⁰⁾が定義した境界の向き(sense of boundary)を含む。すなわち、輪郭線が面を作るとき、反時計回りに輪郭線の向きをとり、穴を開けると、時計回りに向きをとる。

図3は、輪郭線の向きと重複度(巻数の和)の例である。図3aは、上下に交差する8文字の帯を上から射影したもの、図3bは、トーラスを斜めから投影したものである。この場合、裏面($n.e < 0$)を重複度に数えていない。

裏面を隠面処理で無視できない場合、輪郭線は同じ向きに曲面に共有される。図4は蓋のない円柱の例であり、図5はメビウスの帯の例である。図4では、AA'とBB'の輪郭線が同じ向きに共有されている。図5では、AA'の輪郭線が同じ向きに共有されている。書かれている数字は、図3と同じく重複度である。

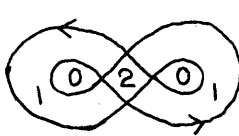


図3 a. 交差する
8文字の帯

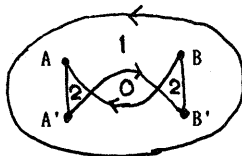


図3 b. 斜めから見た
トーラス

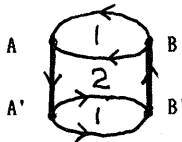


図4 a. 円柱の側面

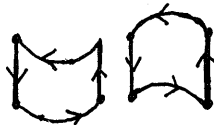


図4 b. 輪郭線の解釈

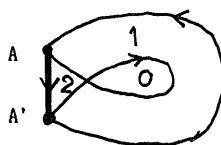


図5 a. メビウスの帯

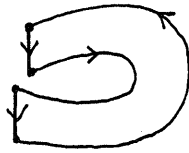


図5 b. 輪郭線の解釈

2.3 輪郭線による隠面処理

輪郭線による隠面処理は、向きをもった輪郭線の演算によって、射影面での重複度をすべて1にする処理といえる。ただし、輪郭線のどの部分を可視にするか判定するには、輪郭線に対応する3次元空間での曲線データをもっている必要があり、対応する曲線については、2.2.1節で議論されている。

2.3.1 重複度を1にする基本演算

隠面処理に関与する射影面での曲線はつぎに分類できる。

(a) 面を定義しない曲線。

(b) 面を片側に定義する閉曲線。

2.2.1節で述べられた輪郭線は、タイプbの曲線である。

また、曲面交差については、すでに処理されタイプbの曲線として加えられているとする。タイプaの曲線としては、例えば、部分的な折り目などを想定している。すべての曲線は向きをもっているとする。タイプaの曲線の向きは、必要に応じて便宜的に定めるものとする。タイプbの曲線の向きは、付随する面にたいし反時計回りとする。

タイプaとbの曲線は、隠面処理によって変形される(演算を受ける)が、他の曲線に隠面処理の影響を与える(演算を引き起こす)のは、タイプbの曲線だけである。

重複度を減らすには、すなわち、巻数の和の値を減らすには、タイプbの曲線にたいし、曲線を制限(削除)するか、逆向きの曲線を加えればよい。曲線の制限は、削除された曲線が見えないとする演算である。逆向きの曲線を加える演算は、可視の曲面の制限に対応する。したがってタイプaの曲線に対しては、曲線の制限だけでよい。

受ける演算は、曲線の制限、逆向きの曲線の追加であるから、点集合としては新しい曲線の作成はない。したがって、すでにある曲線をたどることによって、隠面処理が可能となると期待される。

以下に述べるアルゴリズムにおいて、1つの連結な面に付随するタイプbの曲線は同時に演算を受けるものとして、また、演算を引き起こすものとして、扱うものとする。

WeilerとAtherton⁽³⁾とが提起した隠面処理のアルゴリズムの要点は、つぎのようである。

(1) 演算を受ける曲線が可視か不可視かわくためには、射影面でタイプbの曲線と視線にたいし下側で交差しなければならない。さもなければ、演算を受ける曲線は完全に可視か、不可視である。(この場合には、判別すべき曲線の1点を取り、付録1のように巻数を求め、0なら可視と判定する。プラスの場合には、さらに上下を判定して可視、不可視を定める。)

(2) 下側で交差する場合、演算を引き起こす曲線にたいし右側にする曲線が可視となる。可視の性質は次の下側の交差まで変わらない。したがって、可視曲線がでる交差点のリストをまず作る。リストから1つ出ていく可視曲線を取りだし、閉じるか、不可視となるまで、曲線をたどる。演算を受ける曲線がタイプbのとき、その曲線が面を定義する閉曲線であるためには、可視から不可視が変わるとき、別の曲線に接続しなければならない。この点で出ていく可視の曲線がリストに残っていれば、この曲線に乗り移ればよい。さもなければ、上側にくる演算を引き起こす曲線に移り、リスト上の可視となる交差点か、始点に再び出会うまでそれを逆向きにたどればよい。すべてのリスト上の出ていく可視曲線が使われたとき、隠面処理は完了である。

このWeilerとAthertonとのアルゴリズムは、複雑な閉曲線の

場合には修正を加える必要がある。複雑な閉曲線とは、自分自身が交差するか、巻込み点があるかである。このような複雑な閉曲線では、自分自身で演算を引き起こし、演算を受ける。また、他の曲線にたいしては、上から交差しても、重複度を変えるが、必ずしも不可視を可視に変えない。

修正の方針としては、演算を受ける側のアルゴリズムを変えるか、演算を引き起こす側の輪郭線を重複度が1の面を作る簡単な曲線に還元するかの選択がある。ここでは後者をとる。したがって、タイプbの閉曲線が交差または巻込み点をもつとき、まず自分自身への隠面処理を行い、その射影の面の重複度が1となるようにする。処理された輪郭線が、他の曲線に演算を引き起こすものとする。

2.3.2 自分自身への隠面処理

自分自身への隠面処理にたいしては、2.3.1節のアルゴリズムに次の規則を追加する。ただし、これらの規則では、面を定義する輪郭線を、交差するものを1つのグループとみなし分けるとき、各グループに部分的に可視な輪郭線が少なくとも1つあると仮定している。

【規則A】輪郭線の交差する点において、輪郭線の下側で右に出ていく曲線を、可視曲線の候補としてリストに載せ、リストに残っている限り、それを始点とし2.3.1節のアルゴリズムを開始する。

【規則B】ここで、もとの輪郭線の向きに曲線をたどるときは、演算を受ける側の曲線として考え、それと交差する曲線は演算を引き起こす側とみなす。輪郭線を逆にたどっているときは、演算を引き起こす側に乗り移っているとし、それと交差する曲線は演算を受ける側の曲線とする。

【規則C】演算を受ける側の曲線をたどっているとき、もし、演算を引き起こす側の曲線の下側に左からはいるなら、始点からここまでは不可視とする。(すなわち可視曲線の候補リストから始点を取り除かれる。)

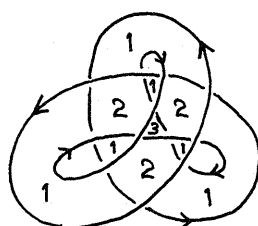


図6 a. 結び目をもつ帯

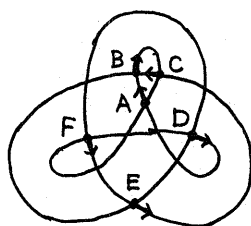


図6 b. 出ていく可視曲線の始点の候補

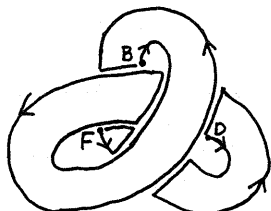


図6 c. 隠面処理の結果

例として、曲面の輪郭線が図6のように自分自身と交わっている場合を考える。図6 aに書かれている数字は、面の重複度(巻数の和)をあらわす。重複度は最高3である。交差点での上下関係を图示するために、下側の曲線にたいしては、交差点で少し間を開けて示す。規則Aより、出ていく可視曲線の候補は、図6 bの点A, B, C, D, E, Fとなる。点Aから出ていく曲線をたどると、点Bで規則Cにより、不可視になり、リストから取り除かれる。残った点からたどれば、結局、図6 cのような隠面処理が行なわれる。

図6の例が示唆するように、自分自身に隠面処理をほどこすと、もとの輪郭線が処理の結果、その向きと逆向きに2度使用されることがある。逆向きの曲線にたいしては隠面を受ける側の面の視線にたいする深さをもっているとし、他の輪郭線に演算をほどこすものとする。

図3 bで与えられたトーラスを処理するためには、上記の規則だけでは不十分で、巻込み点に関する規則が必要になる。このため、巻込み点をまず定義しよう。

輪郭線に沿って面の内側に微小に移動した曲線を考える。このとき、移動幅がどんなに小さくても零でないかぎり、この曲線が射影面で自分自身と交差するなら、輪郭線が巻き込んでいと呼ぶ。輪郭線上の点を取り、どのような小さな近傍に輪郭線を制限しようとも、巻き込みが起きるなら、その点を巻込み点と定義する。

巻込み点は輪郭線を内側に微小に移動した曲線によって分類できる。図7は上下を考慮した曲線の交差の4タイプを示している。曲線の向きは、輪郭線と同じ向きにとられている。このうち、タイプCとタイプDの巻込み点は生じない。なぜなら、巻込み点なら、その曲線を輪郭線に近づけたとき、すなわち、曲線の向きにたいし右側に移動したとき、交差の輪が小さくならねばならない。タイプCとタイプDは反対に大きくなるから起こりえない。

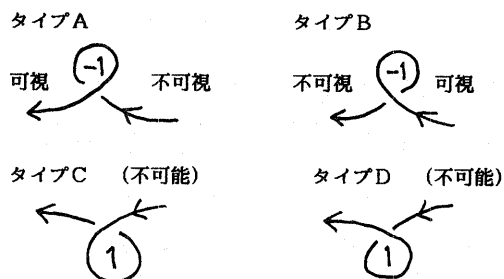


図7. 曲線の交差による巻込み点の分類

図7のタイプAを『出ていく』巻込み点と呼び、タイプBの巻込み点を『入ってくる』巻込み点と呼ぶことにする。この微小に移動した曲線に規則AからCを適用すると、次の巻込み点にたいする規則が得られる。

【規則D】出ていく巻込み点では、輪郭線の向きに出る曲線を、出ていく可視曲線の候補としてリストに加える。もし、

輪郭線の向きに入ってくる曲線があるなら不可視とし、その始点にさかのぼって取り除く。入ってくる巻込み点では、輪郭線の向きに曲線が入ってくるなら、その輪郭線を逆向きにたどる。

図8は規則Dにもとづいて図3bのトーラスを隠面処理したものである。図8aでは、点Aが入ってくる巻込み点で、点Bが出ていく巻込み点になっている。図8bでは、点A'が出ていく巻込み点で、点B'が入ってくる巻込み点になっている。巻込み点の分類を間違えると、全く異なった隠面処理が行われてしまう。

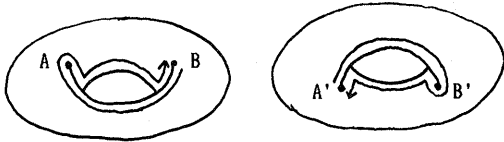


図8a. 上向きのトーラス 図8b. 下向きのトーラス

2.3.3 輪郭線の巻込の発見

ここでは巻込み点の起きる条件と巻込み点の発見方法について述べる。この節で、輪郭線というとき、対応する3次元空間上の曲線をさす。

今、微分可能2次元多様体の内点で輪郭線が巻込み点をもっている場合を考えよう。巻込み点の定義より、輪郭線に沿って内側にとられた曲線は、輪を作って上側の面と下側の面とを連続に結ぶ。この曲線上で面の法線を追跡すれば、つねに視線ベクターと法線ベクターの内積は同符号である。ここで、図9aのように、巻込み点の周りで輪郭線を切断する、視線に平行な面での断面図を考えてみよう。この断面図では、図9bの示唆するように、微分可能2次元多様体がS字型をなしていなければならない。さもなければ、視線ベクターと法線ベクターの内積が、上側の面と下側の面とで同符号でありえない。

これより、重要な結論を得る。常に凸または凹の曲面の内点は、巻込み点になりえない。これは、曲面が有限個の微分可能2次元多様体の組み合わせからなる2次元位相多様体についてもあてはまる。

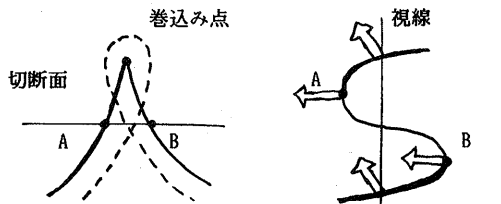


図9a. 射影面上での巻込み点と輪郭線。点線は内側に移動した曲線

図9b. 切断面での曲面の形。矢印は法線の向き。

次に図9aで切断面を巻込み点に近づけていく。このとき、図9bの輪郭線AとBは次第に近づき、巻込み点で連続に接続しなければならない。すなわち、巻込み点では、輪郭線は視線と平行にならなければならない。これより、次の巻込み点の必要条件を得る。

パラメトリック表示では、式(5)の左辺の行列式を $D(s, t)$ とすると、2階連続微分可能2次元多様体の内点からなる巻込み点では、

$$\begin{vmatrix} \partial x / \partial s & \partial D / \partial s \\ \partial x / \partial t & \partial D / \partial t \end{vmatrix} = \begin{vmatrix} \partial y / \partial s & \partial D / \partial s \\ \partial y / \partial t & \partial D / \partial t \end{vmatrix} = 0 \quad (8)$$

が成立する。陰関数表示では、

$$\partial^2 f / \partial z^2 = 0 \quad (9)$$

が成立する。

微分可能2次元多様体の内点では、輪郭線は面を2つに分ける。輪郭線が視線と平行になっても、片側の面から見れば、巻込み点でない。例えば、図9bの輪郭線AとBによって囲まれる曲面から見れば、巻込み点でなくなる。この区別を行うには、輪郭線での法線に着目すればよい。考慮する面の内点では、法線を視点の側にたてる。その面の輪郭線上では、内点の法線と連続になる向きに法線をとる。このとき、式(8)または(9)で与えられる巻込み点の候補にたいし、上側にくる輪郭線ではその法線の向きが面の外側で、下側にくる輪郭線ではその法線の向きが面の内側であるなら、その候補点は巻込み点である。さもなければ、巻込み点でない。巻込み点で、輪郭線が上から下にいくなら、入っていく巻込み点で、下から上にいくなら、出ていく巻込み点である。

図3bのトーラスで、点A, B, A', B'は巻込み点の候補であるが、斜め上から見た場合、可視曲面に対し、点A', B'は巻込み点でなくなる。斜め下から見た場合、点A, Bは巻込み点でなくなる。

2.4 ソート戦略

前節で述べられた輪郭線をたどることによる隠面処理のアルゴリズムは、輪郭線を定義するデータ量Nに対し、 N^2 に比例する計算量を要する。この負担を軽くするのが、この節の狙である。

この処理は、Sutherland⁽¹⁾が述べているように、図形データのもつ同質性(coherence)に着目し、 unnecessaryな計算を省く作業である。輪郭線による隠面処理自体、面を輪郭線に表現し輪郭線の同質性(edge coherence)を利用して、可視、不可視を決めている。したがって、ここで考察されるのは、曲面と曲面の関係の利用である。

図形データの相互の関係は、図10のように見ることができ。1対1の関係でないのは、次の理由による。物体が交差しているときは、曲面を共有しているとし、また、曲面と曲面が接続しているとき、逆向きに曲線を共有していると考えからである。

まず、曲面の集まりが3次元多様体の境界をなすなら、クラインの壺のように曲面が交差しないう限り、向き付け可能で

ある。このとき、法線を外側に向けることにより、視線ベクターと法線ベクターの内積が非負の面だけ処理すればよい⁽⁷⁾。しかしながら、これはデータ量を半分に減らすだけである。

次に、各曲面を直方体で囲み、これによって演算を引き起こす曲面と演算を受ける曲面の組を絞る方法である。Weiler⁽³⁾が提唱した方法は、視線方向でのソートとスクリーン分割の組み合わせである。分子模型のように各曲面の単位が小さいなら、有効となる。Weilerらは、また、接続する曲面を合併し、その輪郭線を単位に隠面処理のアルゴリズムを適用することを示唆している。このとき、前節で述べた輪郭線の自分自身の交差が起きやすくなるので、構成曲面ごとに隠面処理を行って、その後合併するほうが無難である。

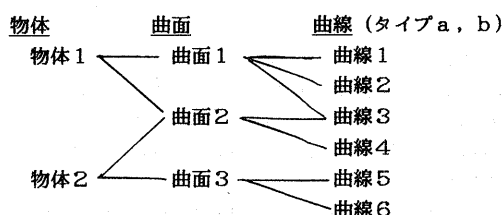


図10. 図形データの関係

3. 簡単なプリミティブへの応用例

3.1 2次曲面

3.1.1 一般の2次曲面

式(1)の陰関数表示の曲面が、 x, y, z の積について2次の関数 $f(x, y, z)$ で与えられるとき、2次曲面と呼ぶ。

2次曲面の輪郭線は非常に簡単である。平行投影なら、式(6)の $\partial f / \partial z$ は1次関数で与えられる。すなわち、式(6)は平面を指定する。したがって、輪郭線は式(1)の指定する曲面と平面の交差曲線となり、その平面上の楕円、双曲線または直線である。射影面上でも楕円、双曲線または直線である。また輪郭線が自分自身と交差することはない。

次に、透視投影を考えてみる。透視投影を行う前の座標系を (x, y, z) 、透視変換後の座標系を (x', y', z') とする。このとき、

$$k = 1 - z'/L$$

$$(x, y, z) = (kx', ky', z') \quad (11)$$

が成立する。ここで、 L は視点の位置である。また、これに対応し、式(6)はつぎに置き換えられる。

$$\frac{\partial f}{\partial z'} = \frac{\partial f}{\partial x} \frac{\partial x}{\partial z'} + \frac{\partial f}{\partial y} \frac{\partial y}{\partial z'} + \frac{\partial f}{\partial z} = 0 \quad (12)$$

式(12)に式(11)を代入し、 $k \partial f / \partial z + 2 f / L$ を求めると、

$$\frac{\partial f}{\partial z} - \frac{1}{L} \left\{ x \frac{\partial f}{\partial x} + y \frac{\partial f}{\partial y} + z \frac{\partial f}{\partial z} - 2f \right\} = 0 \quad (13)$$

となる。式(13)は x, y, z について1次式である。すなわち、

平面を指定する。したがって、透視変換前の座標系で見ると輪郭線は式(1)の曲面と式(13)の平面のなす交差曲線となり、その平面上の楕円、双曲線または直線である。輪郭線が自分自身と交差することはない。

巻込み点は、視点の輪郭線より上側にある限り、平行投影でも射影投影でも生じない。なぜなら、陰関数 f を z または z' の関数と見ると、最大2次で、重複度は2を超えることがないからである。

2次曲面と2次曲面の交差曲線は、一般には4次曲線になる。しかしながら、交差するかどうかの判定式は一般には簡単なものになる。なぜなら、交差するときとしないときの境目では、交差曲線が1点になり、接平面を共有し、方程式系の次数が下がるからである。

3.1.2 球面

球面と平面の交差曲線は円である。したがって、透視変換前の座標系で見ると、輪郭線は円である。球面と球面の交差点は円で簡単に求まる。球面と球面が交差する条件は、球の半径を r_1, r_2 とし、球の中心間の距離を d とすると、

$$r_1 + r_2 \leq d \quad (14)$$

で与えられる。

3.1.3 円柱の側面

円柱の側面と平面との交差曲線は、直線または楕円である。しかしながら、輪郭線はつねに直線だけである。円柱と球面の交差する条件は、円柱の半径を a 、球の半径を r 、円柱の軸と球の中心との最短距離を d とすると、

$$a + r \leq d \quad (15)$$

で与えられる。円柱と円柱の交差する条件は、円柱の半径を

$$a_1 + a_2 \leq d \quad (16)$$

で与えられる。最短距離は、軸へ下ろした垂線の足の長さで求まる。

3.2 パッチ曲面

曲面が分割され、それぞれの曲面がパラメトリック表示で与えられ、接続する境界上で法線を含めて連続なものを、パッチ曲面と呼ぶ。法線を含めて連続にするため、通常、曲面は、パラメータに対して、3次以上の関数になっている。

分割された曲面の隠面処理は、その射影が線分近似の範囲で、Ohno⁽⁴⁾やBonfiglioli⁽⁶⁾が扱っている。両者とも、曲面を四辺形の領域に分割し、その射影パターンを分類し、隠面処理を行っている。多数のパターンが出てくるが、このパターンは、前章で論じた、輪郭線の自分自身の重なり、巻込み点の出現に対応している。

パッチ曲面を正しく曲面として扱うなら、四辺形に限らず、三角形でも輪郭線の自分自身の重なりや巻込み点が出現する。輪郭線は、式(7)で定義されるが、パラメータの高次式となっているため、数値的に解き、2次曲線に近似してやる必要がある。

パッチ曲面の場合、接続している面の集まり(連結な集合)については、曲面の視線方向のソートは不要である。パッチ曲面を輪郭線によって次々とたどることにより、ソートは自然と不要になる。それより、各パッチ曲面がどの境界を通じて他のどのパッチ曲面と接続しているかが、すぐにわかるデータ構造を与えることの方が重要である。リスト構造をもつことによって、これが実現できる。

4. 終りに

輪郭線をたどることによる隠面処理は、射影面での輪郭線の巻数の和と曲面の重複度の関係を利用したものである。この観点から、この方法を曲面に拡張するために、曲面の輪郭線について調べた。

多角形データの場合と曲面が異なるのは、次の3点である。

(1) 視線の変更で輪郭線が動的に変わる。(2) 隠面を作る輪郭線が自分自身と交差することがある。(3) 隠面を作る輪郭線が巻込み点をもつことがある。この(1)のため、輪郭線を求める式をパラメトリック表示の場合と陰関数表示の場合に与えた。また、(2)と(3)のため、新しい規則を作った。また、巻込み点が生じる条件、発見方法を与えた。また、いくつかの曲面に対し輪郭線を具体的に与えた。

したがって、この報告は、曲面の輪郭線による隠面処理の基本的問題ある程度解決したと信じる。しかしながら、実用的なシステムを作るには、輪郭線のデータとしての記述、輪郭線と面との関係の記述など、今後、解決すべき問題がかなり残っている。

謝辞

輪郭線による隠面処理への関心を引き起こし、また適切な参照論文を指摘してくれた東京基礎研究所の青野雅樹氏に感謝したい。彼は、建築システムに、多角形データへの輪郭線による隠面処理を実際に適用し、昨年プログラムを完成させている。

付録1. 閉曲線の内点/外点判別法

与えられた閉曲線に対する内側と外側の点の判別は、円のようにその曲線が陰関数の形で与えられる場合は、直接、判別すべき点の座標を陰関数に代入して、その符号を調べることににより行なえる。領域が上記の領域の均一な集合演算(例えばすべての共通部分集合)で与えられる場合も、その論理演算で行える。凸多角形へのCyrusら⁽¹²⁾のクリッピング法はこの1つの適用例である。

そうでない一般的な場合には、判別はその点の巻数に着目することにより行なえる。

判別すべき点を始点とする半直線を一固定し、巻数カウンタを零にする。閉曲線が半直線と交差するとき、交差点で曲線の向きが時計回りなら-1を、反時計回りなら+1をカウンタに加える。半直線と接する場合には、半直線から離れる点での向きでその値を決める。すべての交差点での値が加えられたとき、巻数がカウンタの値で与えられる。

巻数と内点、外点の関係は、閉曲線の向きの定義に依存する。2.2節の定義では、+1のとき、内点となる。

Surany⁽¹¹⁾の判別法は、上記の判別法を多角形データに適用したもの解釈できる。

判別すべき点が (X_0, Y_0) で、多角形の境界が点列 (X_j, Y_j) のなす辺で与えられているとする。半直線を $X=X_0$ (ここで $Y>Y_0$)ととろう。このとき、辺 (X_j, Y_j) (X_{j+1}, Y_{j+1}) は、 X_j-X_0 と $X_{j+1}-X_0$ とが同符号なら半直線と交差しない。それ以外の場合に、交点 (X_0, Y) を計算し、 $Y>Y_0$ を確かめた上で、辺の向きの値をカウンタに加える。

曲線に適用するときにも、曲線をつくる要素が交差しない場合の簡単な判別条件さえ、工夫すれば速くなる。

参考文献

- (1) I.E.Sutherland, R.F.Sproull, and R.A.Schumacker, 'A characterization of ten hidden-surface algorithms,' ACM Computing Surveys, vol.6, pp1-55 (1974).
- (2) W.M.Newman and R.F.Sproull, 'Principles of Interactive Computer Graphics,' 2nd ed., pp368-369, McGraw-Hill (1979).
- (3) K.Weiler and P.Atherton, 'Hidden-surface removal using polygon area sorting,' Computer Graphics, Vol.11, pp214-222 (1977).
- (4) Y.Ohno, 'A hidden line elimination method for curved surfaces,' Computer-Aided Design, vol.15, pp209-216 (1983).
- (5) R.Duncan, 'Three-space hidden surface removal using boundary traversary logic,' Computer-Aided Design, vol.15, pp217-222 (1983).
- (6) L.Bonfiglioli, 'An algorithm for silhouette of curved surfaces based on graphical relations,' Computer-Aided Design, vol.18, pp95-101 (1986).
- (7) A.Appel, 'The notion of quantitative invisibility and the machine rendering of solids,' Proceedings, ACM National Meeting, pp387-393 (1967).
- (8) R.Abraham, J.E.Marsden, and T.Ratiu, 'Manifolds, Tensor Analysis, and Application,' pp230-250, Addison-Wesley (1983).
- (9) C-C.Hsiung, 'A First Course in Differential Geometry,' pp15-16, pp110-112, John Wiley & Sons, (1981).
- (10) I.E.Sutherland and G.W.Hodgman, 'Reentrant Polygon Clipping,' Comm.ACM, vol.17, pp32-42 (1974).
- (11) A.P.Surany, 'A simple Algorithm for determining a point resides within an arbitrarily shaped polygon,' Fundamental Algorithms for Computer Graphics, edited by R.A.Earnshaw, pp187-193, Springer-Verlag (1985).
- (12) M.Cyrus and J.Beck, 'Generalized two- and three-dimensional clipping,' Computer & Graphics, Vol.3, pp23-28 (1978).