

パラメトリック曲面の交線を求めるための一手法†

小堀 研一^{††} 長田 義弘^{††} 西岡 郁夫^{††}

3次元 CAD のモデリングで最も重要な技術の一つに曲面処理があり、断面形状生成、曲面のトリミング、フィレット曲面生成等の曲面処理はすべて曲面の交線を求める問題に帰着すると言っても過言ではない。本論文では曲面モデルの加工への利用を考慮に入れた精度の良い交線を求めるアルゴリズム、およびそれを応用したフィレット曲面の生成について述べる。従来の手法では曲面を多面体近似するのが一般的であったが、これをそのまま用いると精度、処理速度で問題があった。本文で提案する手法は Quadtree 構造を使って交線の初期点を求め、その後直接曲面データを用いて 2 曲面間のパラメータを同時に変化させて交線を探査するものである。この手法を、既に実用化している当社の 3 次元 CAD/CAM システムに組み込み、実際の電気製品の 3 次元形状の定義に適用し、さらにマシニングセンタでモデルを切削して、その有効性を確認した。

1. ま え が き

設計者が CAD を利用して形状を定義する場合、ディスプレイ上で対話的に計算機内に 3 次元形状を構築していくことが多いので、曲面処理の応答性が重要な要素となり、モデラの操作性にも影響を与える。この曲面処理で基本となるのが 2 曲面間の交線を求める問題であるが、モデラで生成されたコンピュータモデルから直接 NC 加工へも結びつけるためには精度のよい手法が必要である。このため、高速かつ精度の良い交線探索アルゴリズムが要求される。これまで多面体近似計算による交線計算法^{1)~3)}が提案されているが、本論文で扱う曲面に対して精度よく交線を求めるには曲面をかなり細かい多角形に分割する必要があるため実用的でない。また、収束計算による方法^{4), 5)}でも初期値の選び方によって処理速度に難点があった。これを解決するため、交線の存在範囲(干渉している部分)をあらかじめ平面近似により干渉チェックを行って、不必要な計算を減らす方法⁶⁾も提案されているが、精度の高い平面近似を行うと、本来高速に行うべきチェックに時間がかかるという問題がある。そこで筆者らは Quadtree 構造⁶⁾による曲面の存在範囲を示す簡単な直方体を用いて、2 曲面の交線が存在するかどうかを高速に判別し、交線が存在する場合、存在範囲から容易に得られる交線に近い初期点より直接 2 曲面のパラメータを同時に変化させる収束計算に基づく新しい交線探索手法を考案した。これにより、①曲面に対し、

存在範囲を示す直方体をノードとして持つ Quadtree 構造で表現しているので 2 曲面の干渉チェックの処理時間が短く、交線の初期点の探索が容易となる、②交線探索は直接曲面の u, v パラメータを変化させて行うため、交線の精度は曲面の精度を保証することができる。本論文では、2 曲面の交線を求めるアルゴリズムを干渉チェック、収束計算、交線の生成の各ステップに分けて述べ、またこの手法を用いたフィレット曲面の発生方法についても述べる。

2. 曲面の定義

2次元および3次元空間における点をそれぞれ $(a, b) \in R^2, (a, b, c) \in R^3$ で表す。部分集合 $V \subset R^2$ を

$$V = \{(u, v) | 0 \leq u \leq 1, 0 \leq v \leq 1\}, \quad (1)$$

で定義し、これを基準面と言う。今、三つの連続関数 $x(u, v), y(u, v), z(u, v)$ が与えられた時、部分集合 $S \subset R^3$ を

$$S = \{(x(u, v), y(u, v), z(u, v)) | 0 \leq u \leq 1, 0 \leq v \leq 1\}, \quad (2)$$

で定義する。 S を関数 x, y, z で定まるパラメトリック曲面⁷⁾と言う。また、簡単のために、 V から R^3 への関数 P を用いて

$$S = \{P(u, v)\}, \quad (3)$$

と表すこともある。例えば $(0, 0, 0), (50, 0, 50), (100, 0, 0)$ を通る円弧と $(0, 100, 0), (100, 100, 0)$ の直線からなる線織面 S (Ruled Surface) は、

$$x(u, v) = 100 \cdot u \cdot v + 50(1-v)(1 - \cos u\pi),$$

$$y(u, v) = 100 \cdot v,$$

$$z(u, v) = 50(1-v) \sin(u\pi),$$

$$(0 \leq u \leq 1, 0 \leq v \leq 1),$$

なる三つの連続関数により、 $S \doteq \{(x, y, z)\}$ として定

† A Method for Determining the Intersection Curve of Parametric Surfaces by KEN-ICHI KOBORI, YOSHIHIRO NAGATA and IKUO NISHIOKA (Computer Systems Laboratories, Engineering Center, Sharp Corporation).

†† シャープ(株)技術本部コンピュータシステム研究所

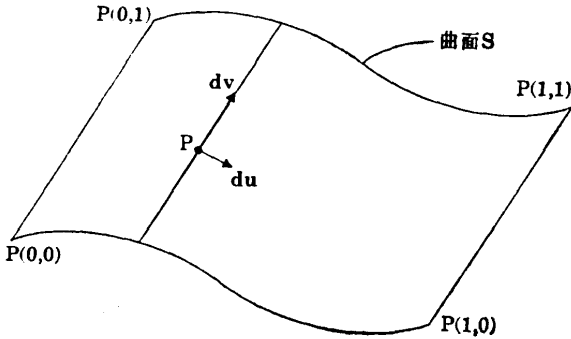


図1 曲面S上の点Pにおける接線ベクトル du, dv
Fig. 1 Tangent vector du, dv at the point P on the surface S.

義される。

また、曲面 S 上の1点 P における u, v 方向の接線ベクトルを

$$\begin{aligned} du &= \partial P(u, v) / \partial u, \\ dv &= \partial P(u, v) / \partial v, \end{aligned} \quad (4)$$

で表す。これを図1に示す。本稿ではこの du, dv が連続である曲面を処理の対象として扱う。次章以降では簡単のためパラメトリック曲面を単に曲面と言う。

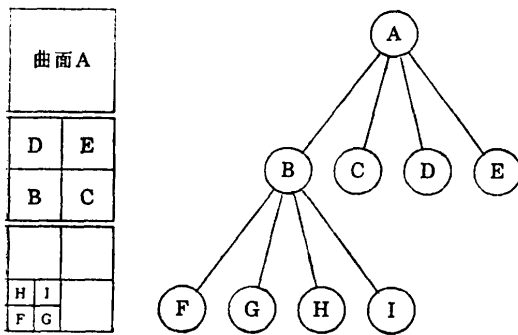


図2 曲面の Quadtree 構造
Fig. 2 Quadtree structure of a surface.

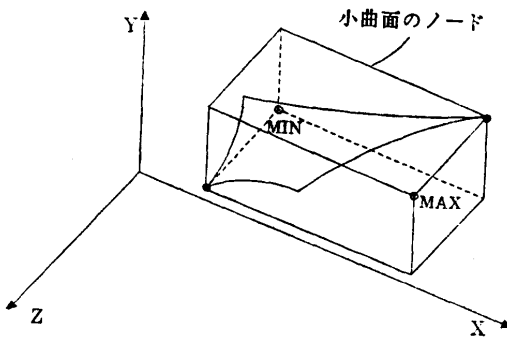


図3 小曲面のノードである直方体
Fig. 3 A min-max enclosing box for a polygon.

3. 曲面の Quadtree 構造

本手法では曲面データに Quadtree データを付加して処理の高速化を図る。Quadtree は3次元座標軸に平行な直方体⁸⁾をノードとして持つ。最下位のノードに対応する直方体は曲面を (u, v) パラメータで $2^n \times 2^n = 4^n$ の格子状に等分割した頂点の座標から求める (n は分割数)。例えば図2の場合、小曲面Fのノードは曲面の頂点 $P(0, 0), P(1/4, 0), P(0, 1/4), P(1/4, 1/4)$ の座標の最小, 最大を使い、その面を含む直方体となる(図3)。また、上位ノードのデータはその下位の四つの直方体ノードの OR をとることにより求める。

4. 二つの曲面の交線を求めるアルゴリズムの概要

二つの曲面 S_1, S_2 が与えられた時、その間の交線は次の四つの手順で求められる。

- Step 1) 二つの曲面 S_1, S_2 よりそれぞれの Quadtree データを作り、そのデータを使って干渉チェックを行うことにより、曲面の交線が存在する最下位ノードを求める。
- Step 2) 干渉チェックで得られたノードに含まれる小曲面の1組(以後ポリゴンペアと言う)より収束計算を行い、交線上の1点(初期点)とそのパラメータ (u, v) を求める。
- Step 3) このパラメータを変化させて、交線上に沿って探索を行いながら交線上の点列を求めていく。
- Step 4) 交線の端点に到達したら、Step 2の初期点から反対方向に Step 3を繰り返し、全体の交線を得る。

5. 二つの曲面の干渉チェック

交線の初期点を決定するために Quadtree データを利用して高速に干渉部を求める手法について述べる。

最初に対象曲面 S_1, S_2 について Quadtree データを作る。次に干渉チェックをこの Quadtree データのノードである直方体が変わるかどうかで判定する。ここで次の記法を用いる。

INT $(N_1, N_2) = F$: ノード N_1 と N_2 は交わりがない。

INT $(N_1, N_2) = T$: ノード N_1 と N_2 は交わる。

S_1, S_2 の最上位ノード A, B の干渉チェック INT

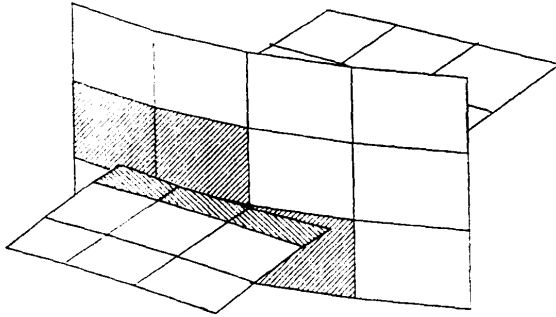


図 4 二つの曲面のポリゴンペア

Fig. 4 Interference check between pairs of polygons of surfaces.

(A, B)=T ならば片方の曲面の Quadtree データのレベルを下げ, INT チェックを続ける. このチェックがその曲面の最下位ノードに達すると, もう一方の曲面の Quadtree のレベルを下げ, 最下位ノードに達した時点で INT チェックが 'T' となっているポリゴンペアを干渉部があるとして記憶する. つまり, 交線の存在する可能性のある領域が見つかったことになる. 図 4 の場合, 斜線の部分で示された小曲面の組が交線の可能性のあるポリゴンペアとして利用される. この方法によれば長方体同士の干渉チェックだけで済むため干渉していない部分が速く判定でき, 処理時間が短くて済む.

この手続きを ITEST としてアルゴリズム⁹⁾に記述する. ここで A, B はそれぞれ曲面 S_1, S_2 の最上位ノードとする.

```

Procedure ITEST (A, B):
  if INT (A, B)=T then
    if A が最下位ノード then
      if B が最下位ノード then
        A, B ペアをストア
      else
        begin
          Bの子ノードを B1, B2, B3, B4 とする;
          ITEST (A, B1);
          ITEST (A, B2);
          ITEST (A, B3);
          ITEST (A, B4);
        end
      else
        begin
          Aの子ノードを A1, A2, A3, A4 とする;
          ITEST (A1, B);
          ITEST (A2, B);
        end
      end
  end

```

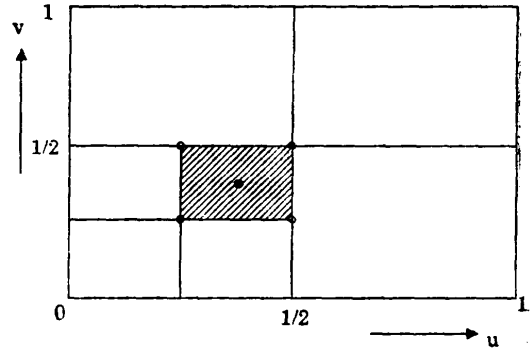


図 5 収束計算の初期点

Fig. 5 An initial point for detecting the curves of intersection.

```

ITEST (A3, B);
ITEST (A4, B)

```

end

end

次にポリゴンペアのそれぞれの4頂点のパラメータの中心 $(u_1, v_1), (u_2, v_2)$ を, 交線を求める初期点として収束計算を行う. 例えば, 図 5 の斜線部のポリゴンの場合, 4頂点の u, v パラメータは $(1/4, 1/4), (2/4, 1/4), (1/4, 2/4), (1/2, 1/2)$ であるから, それらの中心 $(u, v) = (3/8, 3/8)$ が収束計算の初期パラメータとなる. もう一方のポリゴンも同様にして求める.

6. 2曲面からの収束計算アルゴリズム

5章で述べた曲面 S_1 上の交線に近い初期点 $P_1(u_1,$

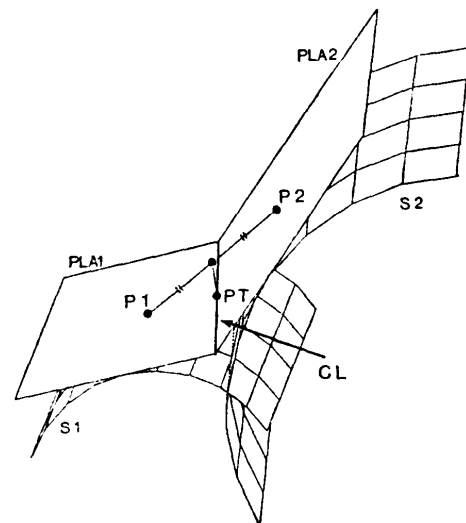


図 6 2平面の交線

Fig. 6 Intersection line CL of planes PLA₁ and PLA₂.

v_1), 曲面 S_2 上の初期点 $P_2(u_2, v_2)$ から収束を開始して $P_1=P_2$ となる交線上の点を求める方法について述べる.

曲面 S_i ($i=1, 2$) の初期パラメータ u, v を u_{in_i}, v_{in_i} とし, 図 6 に示すような $P_i(u_{in_i}, v_{in_i})$ でこの曲面 S_i に接する二つの平面を PLA_i とする. これら平面の法線ベクトルは P_i での u, v 方向の接線ベクトル du_i, dv_i の外積 $du_i \times dv_i$ より求まる. この 2 平面からできる交線を CL とし, P_1 と P_2 を端点とする線分の midpoint から CL 上に垂線を下し, その交点を P_T とする. P_T と P_i よりパラメータ変位量を求め, P_i のパラメータ u, v を更新して新しい P_i とする.

図 7 において S_1 の接平面 PLA_1 上で, P_T と P_1 との変位量は P_T, P_1 の位置ベクトルをそれぞれ $\mathbf{P}_T, \mathbf{P}_1$ とすると $d\mathbf{P} = \mathbf{P}_T - \mathbf{P}_1$ であるからパラメータ変位量 $\Delta u_1, \Delta v_1$ は

$$d\mathbf{P} = \Delta u_1 \times du_1 + \Delta v_1 \times dv_1, \quad (5)$$

の方程式を解けば求まる. P_T と P_2 とのパラメータ変位量 $\Delta u_2, \Delta v_2$ についても同様に接平面 PLA_2 から求まる. この $\Delta u_i, \Delta v_i$ より更新される P_i の u, v パラメータ $u_{in_i'}, v_{in_i}'$ は

$$\begin{aligned} u_{in_i}' &= u_{in_i} + \Delta u_i, \\ v_{in_i}' &= v_{in_i} + \Delta v_i, \end{aligned} \quad (6)$$

となる. このパラメータによる $P_i'(u_{in_i}', v_{in_i}')$ を新し

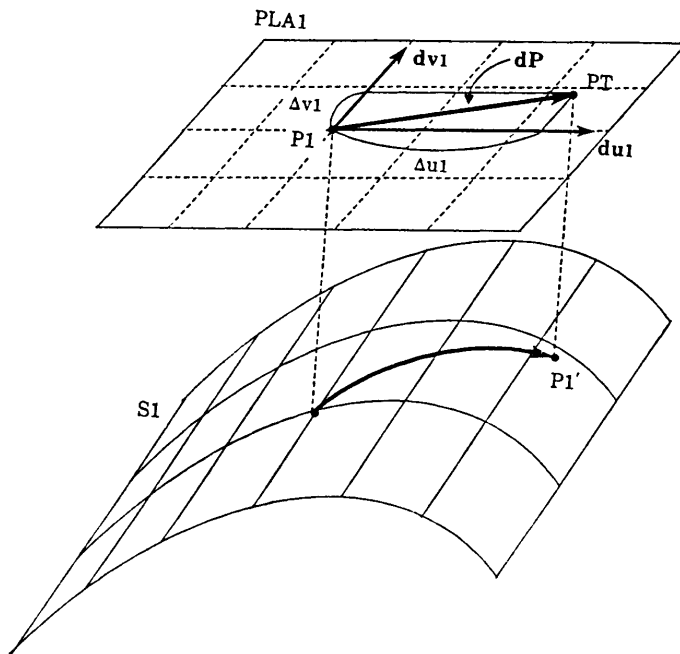


図 7 P_1 の u, v パラメータの更新
Fig. 7 Modification of the parameter u, v of P_1 .

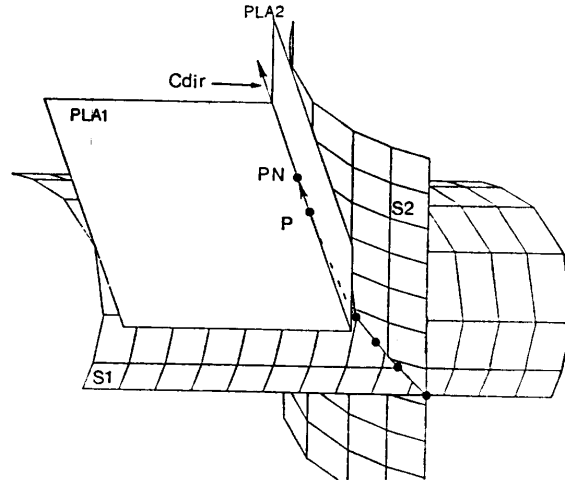


図 8 PLA_1 と PLA_2 の交線方向ベクトル C_{dir}
Fig. 8 Vector C_{dir} along the intersection line of PLA_1 and PLA_2 .

い P_i として上記の手順を繰り返す, もし P_1 と P_2 の距離が許容誤差以下になったら収束したものとし, 交線上の一点が求まることになる.

7. 交線の生成

次に交線上の 1 点から, これに続く交線上の点列を求め, 交線を生成する方法について述べる.

図 8 に示すように, 6 章で求めた交線の初期点 $P = P_1(u_1, v_1) = P_2(u_2, v_2)$ での曲面 S_1, S_2 への接平面を PLA_1, PLA_2 とし, この PLA_1, PLA_2 の交線の単位方向ベクトルを C_{dir} とする. 求める交線の初期点 P における接線ベクトルの向きは C_{dir} と平行である. そこで点 P を $\delta \times C_{dir}$ だけ動かした点を P_N とすると

$$\mathbf{P}_N = \mathbf{P} + \delta \times \mathbf{C}_{dir}, \quad (7)$$

\mathbf{P} : 点 P の位置ベクトル

\mathbf{P}_N : 点 P_N の位置ベクトル

で表される. ここで δ は微小変位量であるが, 曲面の u, v パラメータ密度によって変化させる必要がある. 一般的に複雑な曲面では u, v パラメータを細かくとるので δ を小さくし, u, v パラメータ間隔が大きいく所では δ を大きく設定して, 交線の精度を曲面の精度に合わせて交線を構成する点列が得られるように制御する.

これを実現するため, C_{dir} の du_i 方向,

dv_i 方向のスカラ量 du_i', dv_i' より

$$\delta = C(1/\text{MAX}(|du_i'|, |dv_i'|)), \quad (8)$$

として δ を定める。ここで $i=1, 2$, C は定数。

前章の収束計算の式(5)と同様に

$$dP = P_N - P,$$

$$dP = \Delta u \times du + \Delta v \times dv, \quad (9)$$

の式から $\Delta u, \Delta v$ を求め、次の交線上の点を求めるための収束計算に用いるパラメータ初期値を決める。これをパラメータが曲面の境界に到達するまで繰り返し、交線を生成するための点列を求める。得られた交点列間の精度をあげるため、交点列に対応する曲面の u, v パラメータを等距離に分割した u, v パラメータ列を点列間に発生する。もう一方の曲面に対しても同様に u, v パラメータ列を求め、この各パラメータの組を初期値として6章の収束計算を行って交線上の点とする。この計算はパラメータの組が交線に近いので収束が非常に速い。以上の処理により得られた点列をスプライン曲線で結び、交線を生成する。

交線上の次の点を求める時に、①交線上の前の点を初期値として利用するため、すべての交点を求めて、その点列をソーティングして交線を生成する必要がない、②交線上の次の点は前の点の近傍にあるため、探索が速いという利点がある。

また、2曲面の交差位置によっては交線が2本以上存在することがある。この場合、前章で求めた交線上の1点が、ここで求められた交線上に存在するかどうかを調べる。新たな初期点を P_N 、交線の点列を P_i ($i=1, n, n \geq 2$) とし、 $D_{\max} = \max(P_i, P_{i+1}$ 間の距離), $i=1, n-1$, $D_{\min} = \min(P_N, P_i$ 間の距離), $i=1, n$ を求める。 $D_{\min} \leq D_{\max}$ のとき P_N は交線上にあると判断される。すなわち、交線上の点列間の距離よりも近いところにある P_N は同一交線上の点であるとみなす。この方法で実用的には十分であると考えられる。もし交線上にない点が見つければ、その点を交線の初期点とし、再び、これに続く交線上の点列を求め、交線を生成する。これにより2本以上の交線も探索することができる。

8. 可変フィレット曲面の生成

固定 R のフィレット曲面とは、二つの曲面に接した半径 R の球を、その交線に沿って動かした軌跡の一部であるとする。球の中心の軌跡は、図9のように曲面 S_1, S_2 から、球の半径 R だけオフセットした曲面 S_1', S_2' の交線すなわち交点列となる。

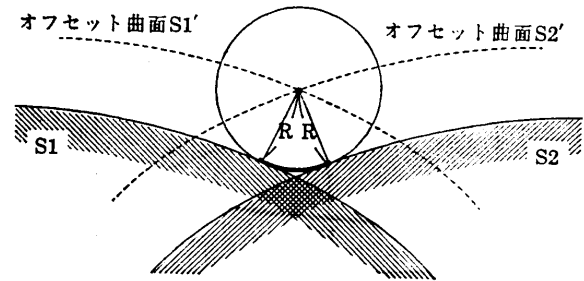


図9 オフセット曲面の交線

Fig. 9 An intersection curve of offset surfaces.

曲面 S_1, S_2 それぞれに関して、オフセット曲面 S_1', S_2' を以下の式

$$S' = S + R \times N(u, v) \quad (0 \leq u \leq 1, 0 \leq v \leq 1), \quad (10)$$

S : パラメトリック曲面

S' : オフセット曲面

$N(u, v)$: 曲面 S 上の点 $P(u, v)$ における単位法線ベクトル

R : オフセット量

より求め、それらオフセット曲面間の交点列を求める。そのオフセット曲面間の交点列と、その交点列に対応する曲面 S_1, S_2 上における (u, v) パラメータ点列を、それぞれ3次式スプライン曲線 $SP_c(t), SP_1(t), SP_2(t)$ ($0 \leq t \leq 1$) で補間する。これによりパラメータ $t=t_i$ の時のフィレット曲面は、中心 $SP_c(t_i)$ 、始点 $SP_1(t_i)$ 、終点 $SP_2(t_i)$ 、半径 R とする円弧として定まる。

ここで曲面のオフセット量である円弧の半径 R を t_i により可変にすると、可変フィレット曲面となる。ここで問題となるのはフィレット曲面の $SP_c(t), SP_1(t), SP_2(t)$ および $R(t)$ をどのように決定するかである。可変フィレット曲面の始点、終点での円弧の半径をそれぞれ R_s, R_e とすると、可変フィレット曲面の中心 $SP_c(t)$ は、フィレットをかける曲面 S_1, S_2 をそれぞれ R_s, R_e だけオフセットさせたときにできる4枚のオフセット曲面の2本の交線間に存在すると仮定し、これらの交線を使ってフィレット曲面の中心軌跡を求める。また、その中心軌跡を曲面 S_1, S_2 に対応させた点列より $SP_1(t), SP_2(t)$ も容易に求まる。以下に中心軌跡を求める詳細な手法について述べる。

まず、曲面 S_1, S_2 を R_s だけオフセットさせてできる曲面同士、 R_e だけオフセットさせてできる曲面同士での交点列を前述の交線を求める手法で計算し、この2本の交点列を曲面 S_1, S_2 上に対応させたときの4本の (u, v) パラメータ点列を、3次式スプラインで補間し、それぞれをパラメータ曲線 $UV_{1s}(t), UV_{1e}(t)$,

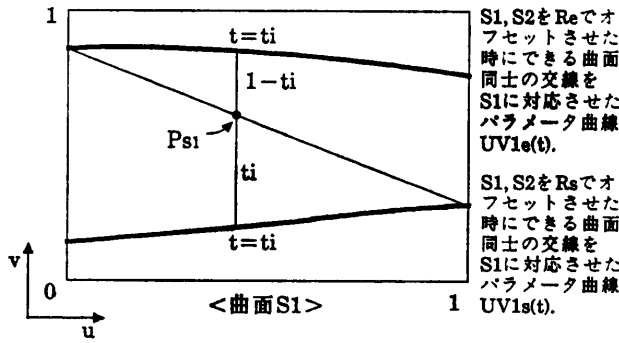


図 10 オフセット面同士の交線を求めるための初期点 P_{s1} の決定

Fig. 10 Calculation of the initial approximation P_{s1} for detecting the intersection curve of two offset surfaces.

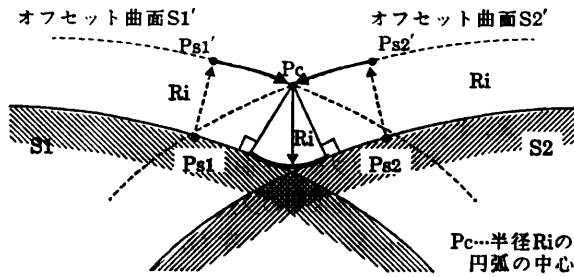


図 11 パラメータ $t=t_i$ の時のフィレット曲面の中心
Fig. 11 Calculation of the center of a fillet at the parameter $t=t_i$.

$UV_{2a}(t), UV_{2c}(t) (0 \leq t \leq 1)$ とする。これら 3 次スプライン曲線とパラメータ t によって、オフセット量 $R(t)$ および中心軌跡を求めるための初期点を決定する。ここで、パラメータ $t=t_i$ のときのオフセット量を R_i 、交点を求めるための初期点として使う S_1, S_2 上の点を P_{s1}, P_{s2} とすると、これらを以下の式で決定する (図 10)。

$$\begin{aligned} P_{s1} &= F_1(t_i) \times UV_{1a}(t_i) + F_2(t_i) \times UV_{1c}(t_i), \\ P_{s2} &= F_1(t_i) \times UV_{2a}(t_i) + F_2(t_i) \times UV_{2c}(t_i), \\ R_i &= F_1(t_i) \times R_a + F_2(t_i) \times R_c, \end{aligned}$$

ここで $F_1(t) = 1-t, F_2(t) = t (0 \leq t \leq 1)$. (11)

この P_{s1}, P_{s2} を R_i だけオフセットした点 P_{s1}', P_{s2}' を初期点として、曲面 S_1, S_2 を R_i だけオフセットした時の曲面間の交点 P_c を、6 章で述べた手法で求める。この様子を図 11 に示す。この交点 P_c は、オフセット量すなわち半径 R_i 時の可変フィレット曲面の中心となる。

この処理をパラメータ t を 0 から 1 まで変化させることにより、可変フィレット曲面の中心軌跡 $SP_c(t)$ が得られる。

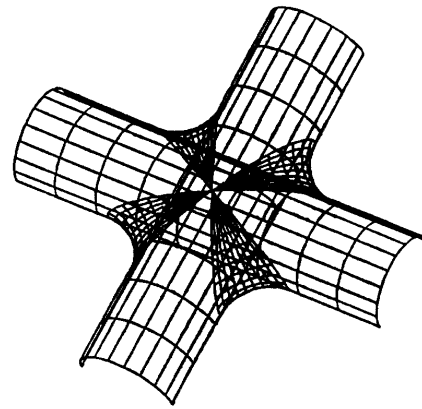


図 12 二つの曲面間のフィレット曲面例

Fig. 12 An example of four fillet surfaces between two surfaces.

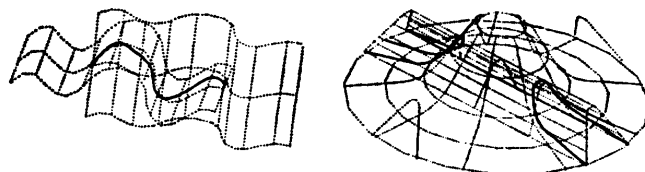
以上の方法からわかるように、中心軌跡を求める際に利用する初期点 P_{s1}' と P_{s2}' は P_c の近傍点としているため、中心 P_c を高速に求めることができる。また、曲面 S_1, S_2 とその間のフィレット曲面が接している点での両者の法線ベクトルが常に一致しているので精度が良い。さらに、式 (11) での関数 $F_1(t), F_2(t)$ を変えることにより、種々の可変 R を持ったフィレット曲面が実現できる。

図 12 はこの手法により求められたフィレット曲面の例である。

9. 実験結果

ここで述べた交線を求める手法を用いて、図 13 の例による計算時間を実測した結果が表 1 である。使用した計算機は DEC 社の VAX11/780 で、すべて FORTRAN 言語で記述されている。

表 1 から明らかなように、全体の処理時間に対する Quadtree のセット、干渉チェックの処理時間が非常に短く、処理の高速性を保証していることがわかる。表 2 は図 13(b) において Quadtree を作成せずに交線求めた場合の各処理時間であり、合計 4.3 秒に対して 33.3 秒の処理時間を要している。表 1 より本手



(a) 交線が 1 本の場合 (b) 交線が 2 本の場合

図 13 二つの曲面間の交線例

Fig. 13 Examples of intersection curves between two surfaces.

表 1 交線生成に要した計算時間

Table 1 Calculation time for obtaining the intersection curve for examples in Fig. 13(a) and 13(b).

	図 13(a)	図 13(b)
Quadtree のセット	0.30	0.30
干渉チェック	0.02	0.02
収束計算	干渉部の全初期点計算	1.60
	点列生成	1.80
複数交線のチェック	0.10	0.10
スプライン生成	0.60	1.20
合計	4.40	4.30

単位: 秒

Quadtree の分割数は $4 \times 4 = 16$

表 2 Quadtree の作成なしに交線生成に要した計算時間 (図 13(b) の例)

Table 2 Calculation time for obtaining the intersection curve without using Quadtree data for examples in Fig. 13(b).

	図 13(b)	
Quadtree のセット	—	
干渉チェック	—	
収束計算	干渉部の全初期点計算	29.90
	点列生成	1.70
複数交線のチェック	0.50	
スプライン生成	1.20	
合計	33.30	

単位: 秒

Quadtree の分割数は $4 \times 4 = 16$

法では Quadtree の作成および干渉チェックには 0.35 秒要しているが、次の処理である干渉部の全初期点を求めるときの収束時間が Quadtree を作成せずに交線を求める場合よりはるかに速くなっている。これは、Quadtree を作成しない場合、初期点を求める際に使う小曲面の組が総当たりの 256 あるのに対して本手法では 24 と減少しているためであり、また干渉チェック処理により、交線上の点を求めるための初期点が交線に近いところに決定されるからである。

交線の精度は式(8)で決定されるが、 C の値を変えた時の処理時間が表3である。 C の値を小さくすると δ は小さくなるので、交線上の点列が多く求まり精度が上がるが、処理時間は増加する。

表 3 式(8)の C を変化させた時の実行時間 (図 13(b) の例)

Table 3 Calculation time for varying the constant C given in equation (8) for the example in Fig. 13(b).

C	1*	2	4
Quadtree のセット	1.00	1.00	1.00
干渉チェック	0.05	0.05	0.05
収束計算	干渉部の全初期点計算	2.90	2.80
	点列生成	2.80	1.80
複数交線のチェック	0.50	0.30	0.20
スプライン生成	1.80	1.20	0.80
合計	9.10	7.20	6.10

単位: 秒

Quadtree の分割数は $8 \times 8 = 64$

* C の各項はこれを 1 として正規化してある。

10. まとめ

対話性を重視する CAD のモデリングでは曲面を多面体近似して処理するのが一般的であり、処理速度も速い。しかしながら、そのデータを加工 (CAM) へ利用するには精度が問題になる。本文で提案した手法によれば、表 1, 2 から明らかなように実用的な応答速度が得られ、かつ曲面の u, v パラメータによって交線を直接求めるため曲面が定義される u, v パラメータの精度で交線、フィレット曲面が求まり、加工への利用が可能になる。本アルゴリズムを当社の 3 次元 CAD/CAM システム¹⁰⁾ に組み込み、図 14 のよう

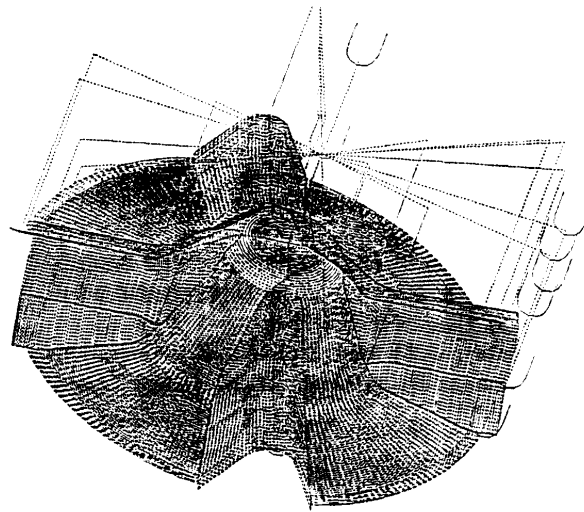


図 14 カットパス発生例

Fig. 14 An example of an NC cutter path.

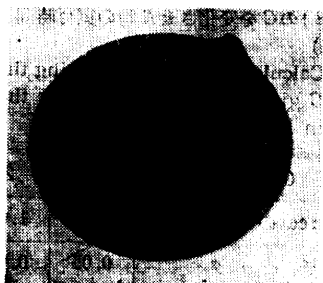


図 15 切削したモデル

Fig. 15 A physical model machined by using the cutter path information.

な NC カットパスを発生させ、実際に NC 工作機によりモデルを切削し(図 15)、その有用性を確認した。

今後の課題としては、曲面の du, dv ベクトルが不連続の場合についても扱えるよう拡張を行い、より汎用的な曲面が扱えるようにアルゴリズムを検討する。また、曲面の複雑さに応じて、最適な Quadtree の分割数を求める手法を検討する。

謝辞 本研究を遂行するに当たり有益な助言をいただいたシャープ精機(株)の CAD グループの方々、および開発に協力いただいたコンピュータシステム研究所 川崎、小笠原両氏に感謝いたします。

参 考 文 献

- 1) 津田, 上西, 矢島: 曲面間相貫曲線の高速計算法, 昭和 56 年度電子通信学会総合全国大会講演論文集, p. 5-315 (1981).
- 2) Carlson, W. E.: An Algorithm and Data Structure for 3D Object Synthesis Using Surface Patch Intersection, *Computer Graphics, Siggraph-ACM*, Vol. 16, No. 3, pp. 255-259 (1982).
- 3) Griffiths, J. F.: A Data-Structure for the Elimination of Hidden Surfaces by Patch Subdivision, *Comput. Aided Des.*, Vol. 7, No. 3, pp. 171-178 (1975).
- 4) 上西, 津田, 赤堀: 断面図作成プログラム SLICE, 昭和 56 年度電子通信学会総合全国大会講演論文集, p. 5-314 (1981).
- 5) 浜川, 川越, 真名垣: 自由曲面を含むソリッド図形処理方式について, 第 29 回情報処理学会全国大会論文集, p. 1755 (1984).
- 6) Hunter, G. M. and Steiglitz, K.: Operations on Images Using Quad Trees, *IEEE Trans.*

Pattern Anal. Machine Intell PAMI-1, No. 2 (April 1979).

- 7) Faux, I. D. and Pratt, M. J.: *Computational Geometry for Design and Manufacture*, pp. 88-107, John Wiley & Sons, New York (1979).
- 8) Hanna, S. L., Abel, J. F. and Greenberg, D. P.: Intersection of Parametric Surfaces by Means of Look-Up Tables, *IEEE Comput. Gr. Appl.*, Vol. 3, No. 7, pp. 39-48 (1983).
- 9) 野崎, 野下 (訳): アルゴリズムの設計と解析 I, pp. 30-35, サイエンス社, 東京 (1977).
- 10) Kobori, K., Nagata, Y., Sato, Y., Jones, K. and Nishioka, I.: Development of a Total 3D CAD/CAM System for Electric Appliances, *Proc. Computer Graphics Tokyo '84*, T4-3 (April 1984).

(昭和 61 年 3 月 31 日受付)

(昭和 61 年 10 月 8 日採録)



小堀 研一 (正会員)

昭和 26 年生。昭和 50 年山梨大学大学院修士課程電子工学専攻修了。

同年シャープ(株)に入社。現在同社コンピュータシステム研究所にて、三次元 CAD/CAM、コンピュー

タ・グラフィックスに関する研究に従事。電子通信学会会員。



長田 義弘 (正会員)

昭和 34 年生。昭和 57 年広島大学工学部第二類(電気系)卒業。同年

シャープ(株)に入社。現在同社コンピュータシステム研究所にて、三次元 CAD に関する研究に従事。



西岡 郁夫 (正会員)

昭和 18 年生。昭和 44 年大阪大学大学院修士課程通信工学専攻修了。

同年シャープ(株)に入社。以来 PCB, LSI, 三次元 CAD/CAM の研究開発に従事。現在、同社コンピ

ュータシステム研究所所長。工学博士。電子通信学会、精密工学会各会員。