# 立体図形の切断面に現れる輪郭線の幾何情報圧縮に関する研究

2V−1

工藤武司 † 品川嘉久 †

東京大学 †

## 1 はじめに

多面体で表される立体図形を平面で切断した際に、断面図には輪郭線を構成する点列が現れる。本論文では,このように点列で表現された曲線の幾何情報を圧縮する方法を提案する。

多くの CAD/CAM システムで採り入れられている NURBS[1] は表現力が高く、曲線形状を一本の曲線として表すことができる。しかし、NURBS で平面上の円を正確に描く場合、最低でも7個の制御点を必要とする。本来、円は中心点の座標と半径を指定すれば決定されるものである。つまり幾何情報圧縮に関しては、NURBS のみを基底とした手法は不適当な場合がある。そこで、基底として NURBS を含めた複数の基底を用意し、曲線情報を効率よくデータ圧縮可能な基底を自動判別する手法を本論文は提案する。

## 2 本手法で用いる基底

断面上の輪郭線、すなわち平面上の曲線を想定し、ここでは各点列の座標値は普通の X-Y 直交座標系を用いている。

- 直線、円、円弧
  円は中心と半径を、また円弧は3点 (1,3 番目の点が端点で、2番目が通過点) をデータとする。

- Cardinal spline 、 Catmull-Rom spline
  両スプライン曲線は与えられた点列間を補間する代表的なもので、この手法では3次の多項式曲線だけを用いる。

- NURBS
  円、円弧以外の一般の円錐曲線を表現するために用いるので、位数を3に限定する。

## 3 輪郭線の幾何情報圧縮アルゴリズム

概要は、まず輪郭線を直線近似 [2] して重要度の高い点列を選別し、次にそれらを元の曲線形状の凹凸の変化が起こる部分でグループ分けを行ない、さらにその各グループ毎に円錐曲線への近似、点間毎にスプライン曲線による補間を順に試みて行くというものである。以下にアルゴリズムを具体的に述べる。

1. (近似曲率計算) 輪郭線を表現する点列集合 $P$ を入力し、各点の近似曲率を求める。近似曲率は隣接する3点を2次多項式曲線で内挿することで求まる。また、ここでは曲率は絶対値を対象とする。

2. (輪郭線の直線近似) 曲率がほぼ0と見なせる点は直線上の点を構成するために取り除き、残りの点列集合を $Q$ とする。$Q$ の点列間の各距離を求め、$Q$ の点 $q_i$ でその両側の点 $q_{i-1}, q_{i+1}$ までの距離が大きく、かつ曲率の大きいものから順に選別し、さらなる新しい点列集合 $R$ を作る。$R$ の要素数は元の輪郭線と圧縮後のそれの誤差及び圧縮率を考慮して適当に決める。（この $R$ の要素同士を直線で結べば、元の輪郭線を直線近似したものが得られる。図2は図1に対する直線近似である。）

3. (変曲点によるグループ分け) 平面上の直線 $ax + by + c = 0$ に関して点 $(x_0, y_0)$ がどちら側にあるかは、$ax_0 + by_0 + c$ の符号により判別できる。ここでは、集合 $R$ の要素 $(x_{i-1}, y_{i-1}), (x_i, y_i)$, $(x_{i+1}, y_{i+1})$ に関して、$(x_i - x_{i-1})(y_{i+1} - y_{i-1}) - (y_i - y_{i+1})(x_{i+1} - x_{i-1})$ の値の符号を用いて $(x_i, y_i)$ における曲率の符号と定義する。(近似曲率の符号は $P$ が整数座標の場合等に信頼できない。) $R$ の要素を走査して、曲率の符号が変化する毎に $R$ の部分集合 $R_j$ を新しく作り、同符号の曲率を持つ点列を $R_j$ に振り分けて行く。ただしそれぞれの $R_j$ を構成する点の内、端点のみは隣接する $R_j$ と重複させ、それらのみはグループの他の要素と逆符号でもよい。

4. (円錐曲線への近似) $\boldsymbol{c}(t)$ を円錐曲線上の点とすると、次式を満たす実数 $w$ と点 $\boldsymbol{b}_0, \boldsymbol{b}_1, \boldsymbol{b}_2$ が存在する。

$$\boldsymbol{c}(t) = \frac{(1-t)^2\boldsymbol{b}_0 + 2w(1-t)t\boldsymbol{b}_1 + t^2\boldsymbol{b}_2}{1 - 2t(1-w) + 2t^2(1-w)}$$

各 $\boldsymbol{b}_i$ は NURBS での制御点、$w$ は $\boldsymbol{b}_1$ に対する重みである。パラメータ $t$ が0から1まで変化するとき、$\boldsymbol{c}(t)$ は点 $\boldsymbol{b}_0, \boldsymbol{b}_2$ を端点とし、制御点で構

---

Geometrical Compression of Contours on Cross Sectional Planes
Takeshi Kudoh† and Yoshihisa Shinagawa† The University of Tokyo†

成される3角形の内部で円錐曲線を描く。曲線形状は$0 < w < 1$で楕円曲線、$w = 1$で放物線、$w > 1$で双曲線となる。また$c(t)$は3点$b_0, b_1, b_2$の重心座標で$c(t) = \tau_0 b_0 + \tau_1 b_1 + \tau_2 b_2$と表現できる。このとき$w = \tau_1/(2\sqrt{\tau_0\tau_2})$の関係がある。各$R_j$の端点は$b_0, b_2$に相当し、それらの接線ベクトルの延長先の交点を求め、$b_1$とする。(もし接線ベクトル同士が平行か、平行に近い場合、あるいは$b_0, b_1, b_2$の作る三角形の外部に$R_j$の端点以外の点が存在してしまう場合には前ステップ3.でさらに$R_j$を分割しておく。)各$R_j$の端点以外の各点は$c(t)$に相当するので、それらの$\tau_i$から$w$を点毎に求める。もしある$R_j$の全$w$がある値に一定範囲内で収まれば、その$R_j$は円錐曲線と見なされる。さらに$\angle b_0 b_2 b_1$と$\angle b_1 b_0 b_2$がほぼ等しく、その余弦が$w$にほぼ等しければ、この$R_j$は円弧にされる。中心と半径が一致する円弧である$R_j$同士が隣接すれば、一つの円弧に変換され、さらに輪郭線全体がこの条件を満たす$R_j$のみで構成される閉曲線であれば、円に変換される。また円、円弧でない円錐曲線部をNURBSのデータに変換し、それらが隣接していれば,同様にまとめたデータ列にする。

5. (スプライン曲線、直線の選別)円錐曲線と見なされなかった$R_j$の要素はスプライン曲線上の点または直線の端点になる。それらの各$R_j$内で隣接する2点$r_i, r_{i+1}$を補間する2種のスプライン曲線の中間点(パラメタ$t$が$r_i$で0, $r_{i+1}$で1をとるとすれば、$t = 0.5$に対応する点)をそれぞれ$r_{i1}, r_{i2}$とする。また$r_i, r_{i+1}$の中点を$r_{ic}$, 元の輪郭線$P$上の点でこれに最も近い点を$p_i$とおく。$r_{i1}, r_{i2}, r_{ic}$それぞれから$p_i$までの距離が最小となる点を選び、その選ばれた点を作る基底($r_{ic}$ならば直線)が$r_i, r_{i+1}$間の基底となる。

6. (スプライン曲線上の省略可能点の除去)隣接する3点$r_{i-1}, r_i, r_{i+1}$が同じ種類のスプラインで補間されるならば、$r_{i-1}, r_{i+1}$を補間する同種のスプライン関数にパラメタ$t = |r_{i-1} - r_i|/(|r_{i-1} - r_i| + |r_i - r_{i+1}|)$を代入して得られる点$r_i'$を求める。$|r_i - r_i'|$がある誤差範囲内であれば、$r_i$は除去される。新しく隣接することになる3点$r_{i-1}, r_{i+1}, r_{i+2}$等に対しても同様である。このような省略可能点が全てなくなるまで、この操作を繰り返す。

## 4 実験結果

図1を上記のアルゴリズムで圧縮した結果を図3に示す。図3の曲線上の大きな点は輪郭線を表現するデータとして残された点である。弧$abc$は円弧化された部分である。

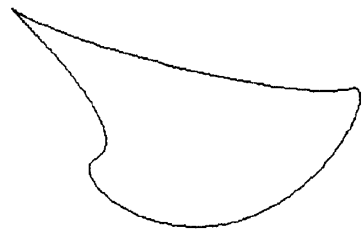

図 1: 輪郭線の例 (点列数 346)

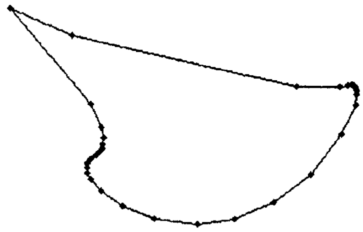

図 2: 図 1 を直線近似した結果 (直線の端点数 36)

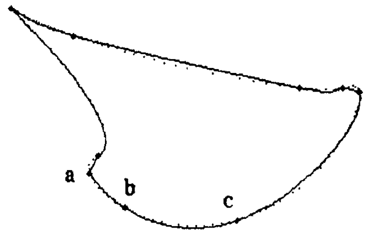


図 3: 図 1 (点線) をデータ圧縮した結果 (実線)

## 5 おわりに

本圧縮手法はスプライン曲線、NURBS等の一般に普及している基底のデータ列への変換でもあるから、局所変形操作等が容易となるという、圧縮以外の面での利点もある。操作性の向上については今後の課題である。

## 参考文献


[1] K.J. Versprille. Computer-Aided Design Applications of the Rational B-spline Approximation Form. PhD thesis, Syracuse University, N.Y, 1975.

[2] Bernd Hamann & Jiann-Liang Chen. "Data point selection for piecewise linear curve approximation" Computer Aided Geometric Design Vol.11, Number 3 (1994) p.289–301.