

拡張 2 次有理 Bézier 曲線による高品位文字フォントの生成とその特徴†

齊 藤 剛†† 穂 坂 衛††

先に報告した拡張 2 次有理 Bézier 曲線による曲線近似法の応用と拡張とを述べる。まず、毛筆などの滑らかな文字の輪郭線を近似することにより、文字の品質を保持したまま、自動的に、少ないセグメント数で高品位文字フォントが作成できることを示す。次いで、先の近似法を拡張し、線図形を記述する点の間の距離が一律でない場合も、指定された誤差範囲で、最少のセグメント数で近似できるようにする。これを用いて、既に直線近似されているフォントを、その品位を上げ、しかも少ない記憶量で曲線近似する方法を示す。最後に、近似曲線に 2 次有理 Bézier 曲線を用いた利点として、座標変換および領域埋めが容易かつ効率的に行えることを示す。

1. はじめに

拡張 2 次有理 Bézier 曲線^{1),2)}を用いて、与えられた点列が密である線図形を、円錐曲線を用いる限り最少のセグメント数で近似する新しい方法を示した^{2),3)}。生成される近似曲線には次の特長がある。(1) Affine 変換が制御点のみで済む。(2) 分割や次数上げ等の手法や性質が知られているので、近似曲線の人手による修正が容易である。(3) 陰関数表現を求めたので、これにより直線との交点が容易に求まる。(4) 近似曲線セグメント同士は接線連続である。

本論文では、以下の 3 点を述べる。

(1) 報告した曲線近似法を、高品位ベクトル文字フォントの作成に応用する。スキャナにより文字を画像として取り込み、その輪郭線に 2 次有理 Bézier 曲線を当てはめ、制御点を保持する。直線を含む円錐曲線を用いる限り最少のセグメント数でフォントが作成できる。これらは、描かれた文字や記号から、新たにアウトラインフォントを作成する場合に用いる。

(2) 曲線近似法を拡張し、既に直線近似された文字や記号を 2 次有理 Bézier 曲線により曲線表現する。プロッタへの出力、特殊な文字や記号のために作られた直線近似によるフォントに対して、その質が高められ、保持すべき座標の数を減少させることができる。また、Affine 変換を施しても品位が保てる。

(3) 2 次有理 Bézier 曲線により表された閉領域を効率的に領域埋めする。また、Affine 変換がその制

御点のみの変換で済む。

アウトラインフォントにおける輪郭線の表現には、直線と円弧⁴⁾、2 次や 3 次の Bézier 曲線およびスプライン^{4)-7),9)}等が利用されている。本論文の方法の特長は、以下の点である。(1) 円錐曲線を用いる限り最少のセグメント数で表現できる。(2) 許容誤差を指定するだけで、近似セグメントが得られる。(3) 高次の方程式を解く必要がない。

2. 拡張 2 次有理 Bézier 曲線の形状と矢高ベクトルの導入

拡張 2 次有理 Bézier 曲線は、従来、正と定義されてきた有理 Bézier 式中の重みを、0 を含む負の領域にまで拡張したものである。これにより、1 つの Bézier 式で表現できる範囲が拡大し、近似曲線として利用できる部分が広げられた。拡張された 2 次有理 Bézier 曲線の諸性質については、他で述べた^{1),2)}。本章では、拡張 2 次有理 Bézier 曲線については、その形状を図示するにとどめ、生成された曲線セグメント列を一樣な量と精度で表現する方法を述べる。

2.1 一般形と曲線形状^{1),2)}

2 次有理 Bézier 曲線は、パラメトリックに次式で表現される。

$$r(t:w) = \frac{(1-t)^2 P_0 + 2t(1-t)w P_1 + t^2 P_2}{(1-t)^2 + 2t(1-t)w + t^2} \quad (2.1)$$

ここで、 $t(0 \leq t \leq 1)$ はパラメータ、 P_0, P_1, P_2 は制御点の位置ベクトル、 w は重みである。この重みが 1 であるときは通常の (有理でない) 2 次 Bézier 式である。各制御点を、一般性を失わずにわかりやすい形にするために、

$$P_0 = (-c, 0), P_1 = (b, h), P_2 = (c, 0) \quad (2.2)$$

† High Quality Outline Fonts by the Extended Rational Quadratic Bézier Curve by TSUYOSHI SAITOH and MAMORU HOSAKA (Department of Electrical Communication Engineering, Faculty of Engineering, Tokyo Denki University).

†† 東京電機大学工学部電気通信工学科

に取る。(2.1)の陰関数表現は,

$$h^2x^2 - 2bhxy + (c^2(1/w^2 - 1) + b^2)y^2 + 2c^2by - c^2h^2 = 0 \quad (2.3)$$

となる(一般的な場合は付録Aに示す)。これは、円錐曲線の式であり、重み w の値により曲線 $r(t:w)$ の形状は、双曲線 ($w^2 > 1$)、放物線 ($w^2 = 1$)、楕円 ($w^2 < 1$) となる(図1参照)。

図1に示したように、負の重みを持つ曲線は、従来からの凸閉包性 (convex hull property) は成り立たない。筆者らが開発した近似法では、計算機により、自動的に、近似曲線を生成する。そこには、凸閉包なる性質を用いないため、この条件を除外することにした。これにより利用できる範囲が広げられ、円錐曲線を用いる限り最長の区分が1つの曲線セグメントで近似できるようになった。

2.2 「矢高ベクトル」の導入

曲線の高さを表す「矢高ベクトル」を導入し、接線連続である曲線セグメント列を一様な量と精度で表現する。

近似しようとする点列の座標値が、 xy 各々 N ビットの整数で表されているとする。この点列を近似する2次有理 Bézier 曲線の第1および第3制御点は、与えられた点であるので N ビット表現できる。しかし、第2制御点は、無限遠の場合を含み、一般には N ビットで表現できない。本節で導入する矢高ベクトルを用いることにより、これができる。

ある曲線セグメント $r(t:w)$ に対して、

$$b = r(0.5:w) - r(0.5:0) \quad (2.4)$$

なるベクトルを、 $r(t:w)$ の「矢高ベクトル」(camber vector) と呼ぶことにする。ここで、

$$r(0.5:0) = (P_0 + P_2)/2 \quad (2.5)$$

であるので、 b は制御辺 P_0P_2 の中点から、曲線上の $t=0.5$ に相当する点に向かうベクトルである(図2)。

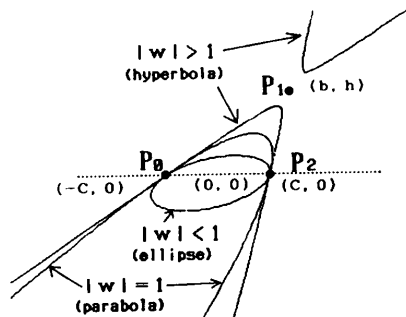


図1 曲線形状と制御点および重みとの関係
Fig. 1 Relations among control points, weights and their curve shapes.

この矢高ベクトル b と重みの関係は以下のとおりである。図3に示すように、 P_0, P_1, P_2 および b が与えられたとする。ここで、

$$Q = b + r(0.5:0) = b + (P_0 + P_2)/2 \quad (2.6)$$

と置くと、3つの位置ベクトル $(P_0 + P_2)/2, Q, P_1$ は同一線上にある。しかも、その間の距離には、

$$|b| : |P_1 - Q| = 1 : w \quad (2.7)$$

なる関係が成り立つ²⁾。これより重み w が求まり、曲線が一意に定まる。

曲線 $r(t:w)$ がその制御辺 P_0P_2 から最も離れる点では、 $r(0.5:w)$ の点である^{1),2)}。 b を各制御点と同様に N ビットの整数で近似表現したとする。この時、元の曲線 $r(t:w)$ との間に、最大 $1/2$ ビット相当の誤差が生じる。これは、後に述べるフォント作成への応用においては、文節画像入力時の誤差と同程度である。

2.3 曲線セグメントを表現する量の最少化

接線連続な曲線セグメント列 $r_i (0 \leq i \leq m)$ では、各々のセグメント ($1 \leq k \leq m-1$) は、その第1制御点と矢高ベクトルの2つのベクトル量のみ保持すればよい。本節では、セグメント r_k の i 番目の制御点を $P_{(k)i}$ 、そのセグメントの矢高ベクトルを b_i と書くことにする。

以下では、セグメント r_{k-1} までは求められているとし、 r_k の制御点と重みが求められることを示す。

$r_k (1 \leq k \leq m-1)$ の第1制御点 $P_{(k)0}$ は与えられる。

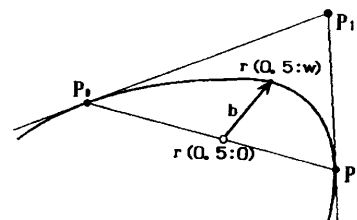


図2 曲線と矢高ベクトルの関係
Fig. 2 Relation between a curve and its "camber vector".

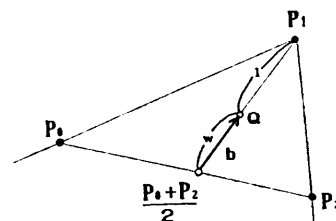


図3 矢高ベクトル、制御辺、重みの幾何学的関係
Fig. 3 Geometric relations among the camber vector (b), control polygon (P_0, P_1, P_2) and weight (w).

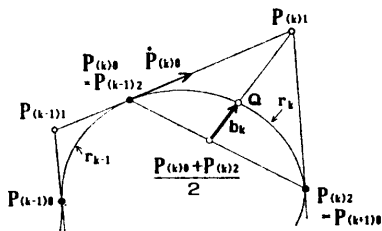


図 4 与えられた $P_{(k)0}$ と b_k から、第2制御点 $P_{(k)1}$ と重み w の求める方法

Fig. 4 A method for obtaining $P_{(k)1}$ and weight, from given control points and its camber vector.

第3制御点 $P_{(k)2}$ は、直後に接続するセグメントの第1制御点 $P_{(k+1)0}$ であり、これも与えられる (図4参照)。したがって、(2.6)に示した Q が求められる。ここで、接線連続である場合、 $P_{(k)0}$ における接線の方法は、直前のセグメントの第2、第3制御点を結ぶ線分 $\overline{P_{(k-1)1}P_{(k-1)2}}$ の方向である (図4の $\dot{P}_{(k)0}$)。したがって $(P_{(k)0} + P_{(k)2})/2$ と Q とを通る直線と、 $P_{(k)0}$ における接線の延長線との交点が第2制御点 $P_{(k)1}$ となる。これにより、3つの制御点が定まる。また、前節(2.7)からその重みが求められる。

以上により、接線連続である曲線セグメント列は、

- (1) 各セグメントの第1制御点 $P_{(k)0}$,
- (2) 各セグメントの矢高ベクトル b_k ,
- (3) 最初のセグメントの第1制御点における接線ベクトル $\dot{P}_{(0)0}$,
- (4) 最後のセグメントの第3制御点 $P_{(m)2}$,

のみ保持すればいいことになる。

接線連続なセグメントの数を L と置くと、それを表すのに必要なベクトル数 T は、

$$T = 2L + 2 \quad (2.8)$$

である。また、そのすべては、近似しようとする点列を表現しているビット数で表すことができる。この具体的な例は、次章で示す。

3. アウトラインフォント作成への応用

本章では、提案した曲線近似法の応用として、アウトラインフォントの自動作成について述べる。文字輪郭線の表現には、直線と円弧⁸⁾、2次や3次の Bézier 曲線およびスプライン^{4)-7), 9)}等が利用されている。本論文では、拡張した2次有理 Bézier 曲線を用い、その制御点を保持する。これにより、直線および円錐曲線を用いる限り、最少のセグメント数で表現できる。

2.3 節の方法により必要な記憶容量を低減できる。

3.1 輪郭線の抽出

まず、スキャナにより入力された原文字画像から、その輪郭線を抽出する方法を述べる。入力画像は、 350×350 のメッシュ状のマトリクスに2値化された形で格納する。スキャナ入力は、市販のソフトウェアを用いた。輪郭線の抽出法として、本論文ではビットシフトおよびビット間論理演算を用いた簡単な方式を採用した。

入力されたマトリクスを B とする。その要素 $b_{i,j}$ ($0 \leq i \leq 349, 0 \leq j \leq 349$) は1ビットである。隣接するビットとの間に次の演算(3.1)を施して、新しい B' を求める。ここで、演算子 $+$, $\cdot$, $\oplus$ は、各々、ビット論理和、論理積、排他的論理和である。

$$b'_{i,j} = b_{i,j} \cdot ((b_{i,j} \oplus b_{i-1,j}) + (b_{i,j} \oplus b_{i+1,j}) + (b_{i,j} \oplus b_{i,j-1}) + (b_{i,j} \oplus b_{i,j+1})) \quad (3.1)$$

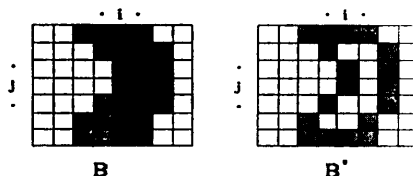


図 5 bit 演算による輪郭線の抽出
Fig. 5 A method of outline (contour) extraction by bit-operations.



図 6 スキャナにより入力した原文字画像
Fig. 6 Original images of characters obtained by a handy scanner.



図 7 検出された輪郭線の例
Fig. 7 An example of extracted outlines of kanji character 「東」 of cursive style.

新しい $b'_{i,j}$ は、原文字画像の輪郭部のみ1となる(図5参照)。このように輪郭部のみ1となったマトリクスの適当な $b'_{i,j}=1$ の点から始め、隣接するビットを検査しながら探索することにより、輪郭線を構成する座標点列を得る。原文字画像(図6)および抽出された輪郭線の1つの例(図7)を示す。

本方法には、例えば、肉厚がメッシュ2区画以下の薄い部分や輪郭部がメッシュ2区画以下に接近している部分などでは正しく輪郭線が抽出できない場合もある。また、入力すべき文字が書かれている紙上の「雑音」部分を輪郭部とみなしてしまう場合がある。これらの修正や除去は、原文字画像入力時に人手により行うことにした。

3.2 フォントの作成結果

3.1 節の方法により抽出した輪郭線を、先に提案した方法により曲線近似し、アウトラインフォントの作成を行った。図6に示した各文字の作成結果を表1に示す。表1では、抽出された輪郭線数、および近似許容誤差を2および2.5とした近似曲線のセグメント数を示す。ここで、近似許容誤差の単位は、入力画像の1メッシュ分に相当する距離である。

また、図8に許容誤差2.5の場合について、各文字の再生結果を示す。図8は、パーソナルコンピュータ

表1 アウトラインフォントの作成例
(輪郭点数と近似セグメント数)

Table 1 The data of generated outline fonts
(number of extracted outline segments
and generated curve segments).

文字	抽出した 輪郭線数	近似曲線セグメント数	
		近似誤差2の場合	近似誤差2.5の場合
C	560	13	11
あ	905	41	37
東	590	35	29
穂	858	63	54

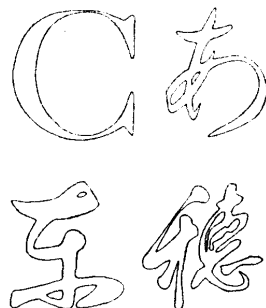


図8 再現文字の例

Fig. 8 Examples of reconstructed characters.

のビデオRAM上にグラフィックスパッケージを用いて350×350ドットで再生しそれをハードコピーしたものである。図8中の凹凸はその際に起こる量子化雑音である。例えば、プロッタ等で描くと滑らかである。図9に草書体「東」における近似セグメントの第1および第3制御点、すなわち、区分の区切りを示す。

3.3 保持情報の量

近似曲線を格納する時、それに必要な記憶容量はできるだけ少ないことが望まれる。寅市らは、利用できるディスクの容量の点から一文字あたり500バイト程度に圧縮する必要があることを述べている⁴⁾。

2.3 節で述べたように、接線連続な曲線セグメントは、最初と最後のセグメントを除き、2つのベクトルのみ保持すればよい。図9および表1よりわかるように、「東」の場合29セグメントである。連続する曲線すなわちループは3つである。したがって、64個の位置ベクトルで表現できることになる。いま、原文字画像を256×256のメッシュで採取したとすれば、1つの位置ベクトルは2バイトで表現できる。「東」は128バイト、セグメント数の比較的多い「穂」が236バイトで表現できる。図10は、約4cm角に書かれた文字を256×256のメッシュで採取し、これを再生したものである。43セグメントで近似され、その保持情報の量は180バイトである。これは、寅市らの示した500バイトに対しては十分な圧縮率である。



図9 各曲線セグメントが表現する区分

Fig. 9 Intervals expressed by each segments.



図10 256×256メッシュで入力文字から作成したフォントの再現例

Fig. 10 A reconstructed font whose original image is printed in 4×4cm and scanned by 256×256 mesh.

4. 曲線近似法の拡張

本章に続く2章では、先に提案した曲線近似法²⁾を拡張し、既に直線近似された文字や記号を2次有理 Bézier 曲線により曲線表現する方法を述べる。

従来、座標変換を必要とする文字や記号は、直線近似されてきた。例えば、プロッタへの出力や特殊な字体の表示のために、既に、多くの文字や記号が直線近似されている。本方法により、曲線部が滑らかに表現できるばかりでなく、保持すべき座標の数を減らすことができる。また、Affine 変換が制御点のみで済み、拡大や回転を行ったとしても、品位が保てる。そのために、先に報告した近似法を拡張する。

4.1 先に提案した曲線近似法の問題点と拡張法

先の近似法での問題点と新たに拡張する点およびその方法を述べる。

問題点と新たに拡張する点

先の近似法は、与えられた点列間の距離のばらつきが少ない場合に、その特長が有効に働く。例えば、3章で示した輪郭線を構成する点間の距離はメッシュ1区分かまたはその $\sqrt{2}$ 倍である。しかし、点の間の距離が一樣でない線図形では、先の近似法の特長である「区分の拡大」が有効に働かず、また、好ましくないうねりが生じることがある。この原因は、(1)各データ点の接線方向を隣接5点から求めたのでは、必ずしも滑らかな曲線のそれとはならないこと、(2)点間の距離が長いとその間に変曲部に相当する部分が含まれてしまうこと、等が主である。特に後者の変曲部を含む区分は1つの2次有理 Bézier では表現できない。このような線図形に適用できるように拡張する。

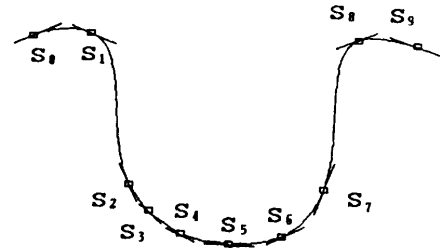
拡張した近似法の概要

拡張した近似法は、次の3ステップからなる(図11参照)。

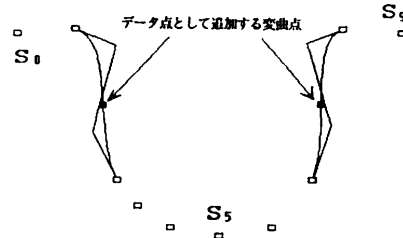
(1) 接線ベクトルの計算: まず、すべての点に対して、その接線ベクトルを求める。そのために、「弾性エネルギーが最少となる曲線」^{11), 12)}を求める(図11(a))。これらは、近似すべき点の数に比例する計算量で計算できる。詳細は次節(a)に示す。

(2) 変曲点の追加: この方法で計算した接線では、その間に変曲点が含まれる場合がある。次に、これを検出し、元の近似すべき点列に追加すると同時にその接線の方向を求める(図11(b))。詳細は、次節(b)で述べる。

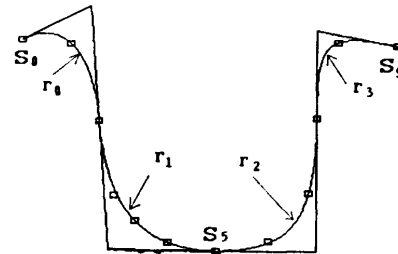
(3) 近似曲線の生成: このようにして作られた点



(a) 与えられた点(□印)、「弾性エネルギーが最少となる曲線」(曲線)、各点の接線方向(直線)



(b) 変曲点をデータ点として追加



(c) 2次有理 Bézier 曲線による近似

図11 拡張した近似法の概要

Fig. 11 Illustrative explanations of curve fitting processes in the extended algorithm.

列を、2次曲線で近似できる最大の区分に分割し、その各々を1つの2次有理 Bézier 曲線で近似する(図11(c): 直線は制御辺である)。生成法は、4.3節に示す。

4.2 接線ベクトルと変曲点の算定法

図11(a)における接線ベクトルの計算法と、(b)の変曲点を求める方法を示す。次節で、近似曲線を得る方法を示す。与えられた点を $S_i (0 \leq i \leq n)$ とする。

(a) 接線ベクトルの計算

各点 S_i の接線ベクトルを q_i とする。近似曲線の形状全体が滑らかであるためには、それらの間に次の関係が成立すればよい¹¹⁾。

$$q_{i-1} + 2k_i(1+k_i)q_i + k_i^2 k_{i+1} q_{i+1} = 3(k_i^2(S_{i+1}-S_i) + (S_i-S_{i-1})), \quad (4.1)$$

および

$$2q_0 + k_1 q_1 = 3(S_1 - S_0),$$

$$q_{n-1} + 2k_n q_n = 3(S_n - S_{n-1}). \quad (4.2)$$

ただし、

$$k_i = |\mathbf{S}_i - \mathbf{S}_{i-1}| / |\mathbf{S}_{i+1} - \mathbf{S}_i|, \\ k_n = 1 \quad (4.3)$$

とする。

(4.1), (4.2) から $n-1$ 個の 3 項方程式が導かれ、データ点の個数のオーダーで、各 $\mathbf{q}_i$ が求まる。

閉曲線の場合、その傾きを定義どおりに解こうとすれば 3 項方程式とはならない。しかし、曲線の終始点の前後を重ね合わせて計算することにより、近似的に求められる。すなわち、与えられた点 $\mathbf{S}_0, \dots, \mathbf{S}_n$ とし、 $\mathbf{S}_0 = \mathbf{S}_n$ である場合、 $\mathbf{S}_{n-k}, \dots, \mathbf{S}_{n-1}$, $\mathbf{S}_0, \mathbf{S}_1, \dots, \mathbf{S}_{n-1}$, $\mathbf{S}_0, \dots, \mathbf{S}_{k-1}$ の $n+2k$ 個を、この順で与えられた点列として計算すればよい。

(b) 変曲点の追加

与えられた点の間に変曲部が生じた場合、その変曲点に相当する点を元のデータに追加し、点の間に変曲部がないようにする。変曲点は以下の方法により求めた。

前述の方法で求めた接線ベクトルを用いて、点 $\mathbf{S}_i$ と点 $\mathbf{S}_{i+1}$ 間を 3 次 Bézier 曲線で補間し、その曲線上の変曲点を求める。まず、点 $\mathbf{S}_i$ と点 $\mathbf{S}_{i+1}$ の間を近似する 3 次 Bézier 曲線は、

$$\mathbf{r}(t) = (1-t)^3 \mathbf{S}_i + 3t(1-t)^2 (\mathbf{S}_i + \mathbf{q}_i/3) \\ + 3t^2(1-t) (\mathbf{S}_{i+1} - \mathbf{q}_i/3) + t^3 \mathbf{S}_{i+1} \quad (4.4)$$

となる。ここで、曲率が 0 となればよいのであるから

$$\dot{\mathbf{r}}(t) \times \ddot{\mathbf{r}}(t) = 0 \quad (4.5)$$

を解けばよい。ここで、 $\dot{\mathbf{r}}(t)$, $\ddot{\mathbf{r}}(t)$ は、 $\mathbf{r}(t)$ の t に関する 1 階および 2 階微分とする。このような点は、 $\dot{\mathbf{r}}(t)$ と $\ddot{\mathbf{r}}(t)$ の方向が一致する点である。これは、付録 B に示すように、 t に関する 2 次方程式を解くことにより求めることができる。傾きは $\dot{\mathbf{r}}(t)$ である。

4.3 近似曲線を得るアルゴリズム

4.2 節の方法により、接線ベクトルが計算されている点列から、それらを近似する曲線セグメントを求めるアルゴリズムを示す。本方法では、与えられたデータ点を 2 分探索法を用いて分割し、1 つのセグメントで近似できる最大の区分を求める。2 分探索であるので、探索の方法は省略する。以下では、初期区分の構成法、および、ある区分に含まれる点列を近似するセグメントの構成法を示す。

(a) 初期区分の構成法

初めに、変曲点で区分分割する。これを初期区分と呼ぶ。これは、2 次曲線を用いるので変曲点は区分の区切りとなるからである。この変曲点とは、(1)4.2

節で述べた追加した変曲点、(2)隣接する前後の点が接線ベクトルに対して異なる側にある点、の 2 つである。初期区分を跨るセグメントはない。したがって、各区分を幾つかの曲線セグメントで近似することになる。また、初期区分内の点では、その接線ベクトルの方向は単調に変化する。

(b) 近似セグメントの構成法

ある区分に含まれる点列（これを、簡単化のために単に $\mathbf{S}'_0, \dots, \mathbf{S}'_m$ と書くことにする）を近似する曲線セグメントの構成法を示す。

まず、3 つの制御点を決定する。 $\mathbf{S}'_0$ を第 1 制御点 $\mathbf{P}_0$, $\mathbf{S}'_m$ を第 3 制御点 $\mathbf{P}_2$ 、および、前節で述べた方法により求めた各々の接線ベクトルの交点を第 2 制御点 $\mathbf{P}_1$ とする制御 3 辺形を作る。

次に、重みを以下の方法で求める。上記のように制御 3 辺形を定めると、それにより表現できる曲線の陰関数表現は、 w を未知数とした $f(x, y; w) = 0$ として求まる（付録 A 参照）。したがって、ある $\mathbf{S}'_k = (x_k, y_k)$ ($1 \leq k \leq m-1$) を通過する曲線の重みは、 $f(x_k, y_k; w_k) = 0$ と置き、 w_k を求めればよい。 $f(x_k, y_k; w_k) = 0$ は、(A.2) からわかるように $w_k^2 = g(x_k, y_k)$ と書ける。制御辺 $\mathbf{P}_0\mathbf{P}_2$ に対して、 $\mathbf{S}'_k$ と $\mathbf{P}_1$ が同じ側にある時は $w_k > 0$ 、異なる場合は $w_k < 0$ とする。次に、各々の点を通過する重みから、 $\mathbf{S}'_1$ から $\mathbf{S}'_{m-1}$ までの各点を近似する 1 本の曲線の重みを求める。そのために、次に示す (4.6) を加重とし、加重平均を求めた。これは、重み w_k を微小に変化させた時、 $\mathbf{S}'_k$ と曲線との距離は e_k に比例した量が変化すると近似できるからである。

$$e_k = \left. \frac{\partial f(x_k, y_k; w)}{\partial w} \right|_{w=w_k} \quad (4.6)$$

このようにして求めた重みを持つ 2 次有理 Bézier 曲線と各点との距離を求め、それが許容範囲か否かを検査する。2 分探索における探索の方向、すなわち、区分を縮小するか拡大するかを、この検査結果により決定する。

5. 直線近似されたフォントの曲線表現と座標変換

既に直線近似され計算機内に格納されているフォントを 4 章で述べた近似法により曲線近似した例を示す。また、これらを座標変換したものを示す。

図 12 (a) は、直線近似されている文字である。□印は、直線近似のための座標点である。(b) は、各々

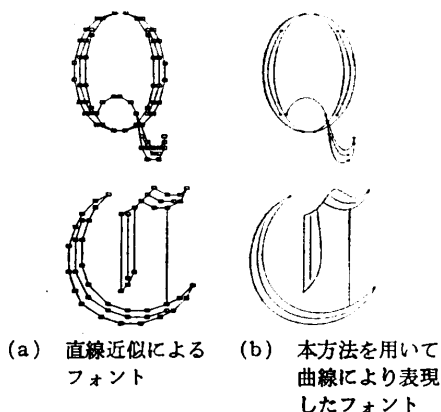


図 12 直線近似されたフォントから曲線近似により品位を高めたフォントの作成
Fig. 12 Generated high quality fonts from their polyline representation.



図 13 「IPSJ」の座標変換の例
Fig. 13 Examples of Affine transformations.

を曲線近似した例である。文字「Q」の作成フォントには、セグメントの区切りに●印を付けた。

品位が高められたばかりでなく、必要な記憶容量も減少された。文字「Q」の直線近似には、69点を要している。本方法により13個のセグメントで近似された。第2章で述べた方法によると42ベクトルで表現され、保持すべき座標の数は約61%に低減された。また、文字「C」の直線近似には59点を要しているが、13セグメントで曲線近似できる。

図13は、ゴシック文字「IPSJ」から作成したフォントとその座標変換の例である。

6. 2次有理 Bézier 曲線により表現された閉領域の領域埋め

2次有理 Bézier 曲線により表現される閉領域は、その領域埋めが容易にできることを示す。

多边形により表現された閉領域の領域埋め法は種々報告されている¹⁰⁾。領域埋めを行うためには、その領域の境界と直線の交点を求める必要がある。これを容易に繰返しなしで求められるためには、その交点を求める式が2次以下でなければならない。

筆者らは、パラメトリックに表現された拡張2次有理 Bézier 曲線の陰関数表現を求めた。これを用いることにより、曲線と直線との交点は2次方程式を解くことのみで求められる。

6.1 領域埋めアルゴリズム

ここで、 $L = \{L_0, L_1, L_2, \dots, L_m\}$ を領域埋めに用いる直線の集合、 $r_i (0 \leq i \leq n)$ を閉領域を記述する2次有理 Bézier 曲線群とする。アルゴリズムの概要を図14に示す。変数 Cross_Point は、要素が交点である集合型変数とする。

以下では、図14に示す領域埋めの概要を、図15を例にとり、その流れに従って述べる。

図15は、 x 軸に平行する直線 $L_k: y=c_k$ の場合を示す。図14における[4]の繰返しにより、直線 L_k とすべての $r_i (0 \leq i \leq n)$ との交点を求め、それを[6]で、 x 成分の値で整列化する。交点数を h として、整列化後の交点を $a_j (j=0..h-1)$ と書く。昇順でも降順であってもよい。図15は昇順の場合を示す。次に[7]で、 $j=0..(h-2)/2$ とし、線分 (a_{2j}, a_{2j+1}) を描く。これにより、直線 L_k の、閉領域内の部分が描かれたことになる。これら一連の処理を、各 L_k について行えばよい。また、 L_k が x 軸に平行でない場合、

```
[1] for  $k := 0$  to  $m$  do
[2]   begin
[3]     Cross_Point := 空;
[4]     for  $i := 0$  to  $n$  do
[5]       Cross_Point :=
         Cross_Point  $\cup$  (直線  $L_k$  と曲線  $r_i$  との交点)
[6]     sort(Cross_Point);
[7]     for  $j := 0$  to (集合 Cross_Point の要素数) - 1
       step 2 do
[8]       線分 (Cross_Point( $j$ ), Cross_Point( $j+1$ ))
         を描く;
[9]   end;
```

図 14 直線による領域埋めアルゴリズム
Fig. 14 An algorithm for filling an area by line segments.

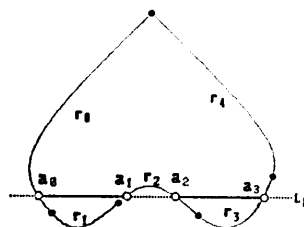


図 15 交点と領域内線分 (●は曲線の第1, 3制御点である)
Fig. 15 Intersection points between a line and curves. (● denotes 1st or 3rd control points of each curve segment.)

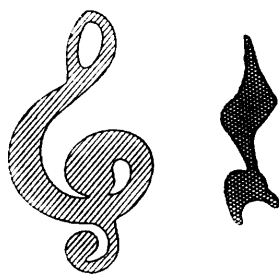


図 16 斜めの平行直線による領域埋めの例
Fig. 16 Examples of filled area.

曲線群を回転させるか、または、〔6〕で L_* の方向に添って整列化すればよい。

交点の集合を求める過程〔5〕で、単にその解の集合を求めるのみでは不都合が生じる場合がある。領域埋めに用いる直線と曲線が接する場合や交点が曲線の制御点と一致する時などがある。しかし、例えば、セグメントの端点以外での重複解の場合は交点に加えない、終端点で重複解の場合は交点とするが始端点の場合は加えない等の対策を施せばよい。

6.2 領域埋めの例

図 16 は、回転と領域埋めを組み合わせた例である。回転は、図 13 と同様に制御点への変換のみで行える。次いで、 x 軸に平行な直線と各々の近似セグメントとの交点を求める。そして、交点を逆に回転させ直線を描く。 y を適当な幅で変化させることにより、図 16 が描かれる。

7. おわりに

本論文では、2 次有理 Bézier 曲線を用いた曲線近似法の応用として、アウトラインフォント作成について述べた。これを自動的に行うシステムをパソコン上で開発した。3.1 節で述べた「雑音の消去」以外は、人が許容誤差を指定するだけで、入力された原文字画像から近似曲線を求めるまでは自動化されている。圧縮率および文字の質の点からも、本方法の有効性が検証された。文字ばかりでなく記号やマークの表現にも応用できる。また、既に直線近似された文字や記号を、曲線により近似表現する方法についても述べた。これにより、曲線部が滑らかに表現できるばかりでなく、保持すべき情報の量を低減できることを示した。制御点を保存したことにより、その品位を保ったまま Affine 座標変換が可能であり、また、領域埋めができることを示した。

筆者らは、タブレット入力による手書き線図形を本方式で近似し、保持情報の圧縮も試みた。これらは、

別の機会に報告したい。

2 次有理 Bézier 曲線の表示には、割算を含む実数計算が必要となる。これらを少ない誤差で高速に行うアルゴリズムの開発は今後の課題である。

謝辞 本研究の遂行に際し、本学研究生 久志本琢也氏から有用な助言を頂きました。また、大学院生 佐藤龍氏（現（株）QUICK）には、各種の表示出力プログラムを作成して頂きました。ここに感謝いたします。また、日頃、ご協力頂きます本学大学院生 権田秀夫氏に感謝します。

参 考 文 献

- 1) 穂坂, 久志本, 権田: 曲線曲面の接続と干渉(続), 情報処理学会 G&CAD シンポジウム, pp. 121-132 (1988).
- 2) 斉藤, 穂坂: 拡張した 2 次有理 Bézier 曲線の性質とその曲線近似法への応用, 情報処理学会論文誌, Vol. 31, No. 1, pp. 33-41 (1990).
- 3) 斉藤, 穂坂: 文字輪郭線の 2 次有理ベジェ式による近似, 信学全大, D-432 (1989).
- 4) 寅市, 関田, 森: 高品位文字フォントの自動圧縮, 信学論, Vol. J70-D, No. 6, pp. 1164-1172 (1987).
- 5) Murayama, N.: A Polynomial for Automatic Contour Vector Coding, *Trans. of the IEICE*, E 72-5 (1989).
- 6) Plass, M. and Stone, M.: Curve-Fitting with Piecewise Parametric Cubics, *Comput. Gr.*, Vol. 17, No. 3, pp. 229-239 (1983).
- 7) 曾我, 豊田: 小型 DTP システム用漢字書体生成支援システムの構築, 情報処理学会 G&CAD シンポジウム, pp. 25-34 (1988).
- 8) 山崎, 井村: 文字輪郭線の円弧と直線による近似, 情報処理学会論文誌, Vol. 26, No. 7, pp. 726-732 (1985).
- 9) 山田, 佐藤: 線分近似表現から 3 次の Bézier 曲線表現への変換手法, 情報処理学会 G&CAD 研究会, 38-2 (1989).
- 10) Herman, I. and Revczky, J.: New Method for Improving the GKS Fill Area Primitive, *Comput. Gr. Forum*, Vol. 6, pp. 195-202 (1986).
- 11) 穂坂: 曲線, 曲面の合成および平滑化理論, 情報処理, Vol. 10, No. 3, pp. 121-131 (1969).
- 12) Farin, G.: *Curves and Surfaces for Computer Aided Geometric Design—A Practical Guide*, Academic Press, Inc. (1988).

付 録 A

2 次有理 Bézier 曲線の 3 つの制御点を

$$\mathbf{P}_0 = (x_0, y_0), \mathbf{P}_1 = (x_1, y_1), \mathbf{P}_2 = (x_2, y_2)$$

(A.1)

にとる。(2.1)で, $u=t/(1-t)$ と置き, x, y 各々に關する u の2次式を2つ得る. これを連立させ, u を消去する.(2.1)の陰関数表現を次式として得る.

$$\begin{aligned} f(x, y; w) &= x^2(y_0 - y_2)^2 + y^2(x_2 - y_0)^2 \\ &+ 2xy(y_0 - y_2)(x_2 - x_0) \\ &+ 2x(y_0 - y_2)(x_0y_2 - x_2y_0) \\ &+ 2y(x_2 - x_0)(x_0y_2 - x_2y_0) \\ &+ (x_0y_2 - x_2y_0)^2 \\ &- 4w^2[x^2\{(y_1 - y_0)(y_2 - y_1)\} \\ &+ y^2\{(x_0 - x_1)(x_1 - x_2)\} \\ &+ xy\{(x_0 - x_1)(y_2 - y_1) + (y_1 - y_0)(x_1 - x_2)\} \\ &+ x\{(y_2 - y_1)(x_1y_0 - x_0y_1)\} \\ &+ (y_1 - y_0)(x_2y_1 - x_1y_2)\} \\ &+ y\{(x_1 - x_2)(x_1y_0 - x_0y_1)\} \\ &+ (x_0 - x_1)(x_2y_1 - x_1y_2)\} \\ &+ (x_1y_0 - x_0y_1)(x_2y_1 - x_1y_2)] = 0. \quad (\text{A.2}) \end{aligned}$$

付 録 B

3次 Bézier 曲線を

$$\begin{aligned} r(t) &= (1-t)^3P_0 + 3t(1-t)^2P_1 + 3t^2(1-t)P_2 + t^3P_3 \\ & \quad (\text{B.1}) \end{aligned}$$

とする. この時,

$$\begin{aligned} \dot{r}(t) &= t^2\{-3P_0 + 9P_1 - 9P_2 + 3P_3\} \\ &+ t\{6P_0 - 12P_1 + 6P_2\} + \{-3P_0 + 3P_1\}, \\ & \quad (\text{B.2}) \end{aligned}$$

$$\begin{aligned} \ddot{r}(t) &= 2t\{-3P_0 + 9P_1 - 9P_2 + 3P_3\} \\ &+ \{6P_0 - 12P_1 + 6P_2\} \quad (\text{B.3}) \end{aligned}$$

である. ここで,

$$\begin{aligned} a &= -3P_0 + 9P_1 - 9P_2 + 3P_3 \\ b &= 6P_0 - 12P_1 + 6P_2 \\ c &= -3P_0 + 3P_1 \quad (\text{B.4}) \end{aligned}$$

と置き, 例えば, a の x 成分を a_x と書く. $\dot{r}(t)$ と $\ddot{r}(t)$ と方向が一致するためには, 以下が成り立てばよい.

$$\frac{a_x t^2 + b_x t + c_x}{a_y t^2 + b_y t + c_y} = \frac{2a_x t + b_x}{2a_y t + b_y} \quad (\text{B.5})$$

これを整理すると,

$$t^2(a_y b_x - a_x b_y) + 2t(a_y c_x - a_x c_y) + (b_y c_x - b_x c_y) = 0 \quad (\text{B.6})$$

となり, 解析的に解ける.

(平成元年9月29日受付)

(平成2年1月16日採録)



斉藤 剛 (正会員)

昭和25年生. 昭和48年東京電機大学工学部電気通信工学科卒業. 昭和51年東京電機大学大学院修士課程修了. 昭和54年4月, 工学部電気通信工学科助手. 現在, グラフィックス, CAD に関する研究に従事. 電子情報通信学会, 日本 ME 学会など各会員.



穂坂 術 (名誉会員)

大正9年生. 昭和17年9月東京大学工学部航空学科卒業. 海軍, 運輸省, 国鉄を経て, 昭和34年10月より東京大学教授, 50年4月より東京工業大学教授兼任, 昭和56年3月東京大学, 東京工業大学定年. 工学博士. 東京大学名誉教授. 現, 東京電機大学教授, 当学会前会長, 調査研究運営委員会委員長. 学会論文賞(当学会2編, 機械学会1編), 当学会功績賞, 紫綬褒章, 科学技術庁長官賞など. 昭和28年からコンピュータに関心を持ち実時間情報システム, グラフィックス, CAD などとその初期から取り扱う.