3次スプライン補間のためのデータ選択の一手法†

原田 耕一††

3次スプライン関数を用いて有限個のデータを補間するとき，データ点の配置がいかなる場合に，データに忠実な補間が行えるかの検討をし，最適データ点列に関する定式化を行う．この目的のために de Boor によって提案された折線近似に関する理論を応用し，3次スプライン補間という見地からの最適な点列に関する定義，および与えられた点列の評価法を与える．さらに適用例として，最適データ点列の生成，データ点列の評価，およびデータ追加を実行し，提案手法の評価を行う．本論文で得られた結果は，デザイナの求めたい曲線，曲面，あるいは，データを与える源である信号を処理するのに用いることができる．

1. まえがき

実験または観測によって得られたデータを計算機によって自動的に処理するための LA (Laboratory Automation) と CAGD (Computer Aided Geometric Design) を比べると，与えられた有限個のデータを滑らかな曲線で補間する点において両者は同じであるが，用いられる補間手法には本質的な違いがある．

前者において，データ点を計算機に与えた後の処理は一意的になされ，データ点を処理中に変更することはまれである．したがって，LA 分野において用いられる信号のモデルとしての曲面（曲線）創成のための補間関数は，データ座標以外の入力（例えば接線ベクトルのような）のない3次スプラインに代表される[1]．

これに対し，後者では，人間と計算機とのインタラクティブな作業によって最終的な関数を決定するという立場をとる．したがって，Bezier 曲線やBスプラインといった制御変数を含む補間曲線が広く用いられている[2]．

いずれの分野においても，得られる補間関数が満足できるものかどうかを決定する上で，データ点の選び方が最終結果に大きな影響を与える．特に，LA 分野における補間においては，データ点が与えられると，用いる補間手法に依存して，最終的な補間関数の形状が一意に決定されるので，データ点の選び方は極めて重要である．

従来，補間手法の研究は，与えられたデータ点に対し，いかに視覚的に自然な曲線を生成するかという立場でなされてきた[3]．しかし，与えられた点列（その点数と配置）が，用いる補間手法に適したものであるかどうか，言い替えると，点列の配置が適正であるかを指摘したり，また，与えられたデータの一部が，用いる補間関数の性質によっては不足していて，満足な補間関数が得られないという警告を与えるような研究はほとんどなされていない．データ点数の過不足について論じた数少ない研究の中に佐藤による成果[4]があるが，ここではデータ点数の議論に研究の目的があり，データ点の配置に関する検討はなされていない．

ところが，実際のデータ点選択についてみると，GAGD においては，オペレータの経験と勘にもとづいて適切と思われるところに設定される．また，実験・観測データの測定場所は，一般に，後の補間処理とは無関係に選択され，場合によっては，物理的に計測が不可能もしくは非常に困難な場所すら存在することがある．このようにして与えられるデータを補間しても，得られる補間曲線が，デザインあるいは計測結果処理の目的を十分反映しているという保証はない．

このような問題に対処するため，本論文では，3次スプライン補間関数を取り上げ，この補間手法に適するデータ点の配列についての検討を行う．具体的には，de Boor によって提案された折線近似に関する理論[5]を3次スプラインに適用し，この補間に適したデータ点列を定式化する．これを利用することによって，最適データ点列の生成，データ点列の評価，およびデータ追加の是非を与えることができ，上述の CAGD および計測結果の処理などに役立てることができる．

2. 3次スプライン補間関数の概要

3次スプライン補間関数については広く研究が行われ，多数の著者（例えば文献 6)）がある．ここでは，後述の最適データ点列に関する指標導出の説明を容易にするために必要な定義をしておく．

† A Data Selection Method for Cubic Spline Interpolation by KOICHI HARADA (Faculty of Integrated Arts and Sciences, Hiroshima University).
†† 広島大学総合科学部

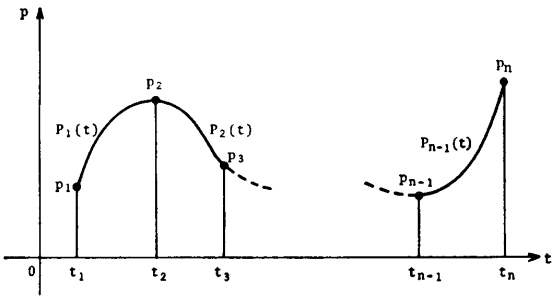

図 1 3次スプライン補間関数
Fig. 1 Cubic spline interpolants.

3次スプライン補間関数は，データ点間ごとに区分的に計算される（**図1**参照）．いま，(t_1, p_1), (t_2, p_2), …, (t_n, p_n) なる n 個のデータ点列が与えられたものとする．ここで，(t_i, p_i); $i=1,2,\cdots,n$ はそれぞれ独立変数および従属変数を表している．隣接する2点のデータ (t_i, p_i), (t_{i+1}, p_{i+1}) $(i=1,2,\cdots,n-1)$ の間はセグメントと呼ばれ，3次スプライン関数により次のように補間される．

$$P_i(u)=B_1+B_2u+B_3u^2+B_4u^3,$$
$$0\leqq u\leqq L_i,\ L_i=t_{i+1}-t_i. \qquad (1)$$

ここで，$u=t-t_i$ である．係数 B_1, B_2, B_3, B_4 はセグメントの接続点において $P_i(0)$, $P_i'(0)$, $P_i(L_i)$, $P_i'(L_i)$ を与えることによって決定される．なお，$'$ は独立変数に関する1階導関数を表している．また，本論文においては，データ点は各セグメントの接続点に常に一致し，B-スプラインにおけるように，セグメントの接続点（節点[7]）とデータの位置が異なる場合は含めないものとする．

(1)式を，これら四つの値を用いて書き換えれば，次式が得られる．

$$\begin{aligned}P_i(u)=&P_i(0)+P_i'(0)u+[3\{P_i(L_i)-P_i(0)\}\\&-L_i\{2P_i'(0)+P_i'(L_i)\}]u^2/L_i^2\\&+[2\{P_i(0)-P_i(L_i)\}+L_i\{P_i'(0)\\&+P_i'(L_i)\}]u^3/L_i^3. \qquad (2)\end{aligned}$$

(2)式において，$P_i(0)$, $P_i(L_i)$ はデータとして与えられるので，$P_i'(0)$, $P_i'(L_i)$ を求めれば3次スプライン関数を決定することができる．これらの1階導関数の値は，各セグメントの接続点において，関数の独立変数に関する2階導関数が連続という条件と，データ点列の両端点，すなわち (t_1, p_1), (t_n, p_n) において課せられる条件によって一意に決定することができる．

後者の条件としては次のようなものがある[6]．(i) 強制スプライン：$P_1'(0)$ および $P_{n-1}'(L_{n-1})$ を与える．(ii) 自然スプライン：$P_1''(0)=0$, $P_{n-1}''(L_{n-1})=0$ とする．(iii) 周期スプライン：曲線の両端点において，2階導関数まで一致する．(iii) は閉曲線データに対して有用であるが，一般性の高い開曲線は (i), (ii) を用いて表現される．また，(ii) の条件によって得られるスプラインにおいては，$\{P''(t)\}^2$ の $[t_1, t_n]$ 上での積分（歪エネルギーと呼ばれる）が最小となることが知られており[8]，スプライン曲線の定義そのものである歪エネルギー最小曲線の概念に一致し，スプラインを計算する際の余分な入力パラメータをなくすことができる．しかし，両端点の条件が異なっても，得られるスプライン曲線の形状は，両端点から3点程度離れれば，曲線の形状にほとんど差異が認められなくなることが知られている[9]．

周知のように，自然スプラインのデータ点上での1階導関数 $(P_i'\ (i=1,2,\cdots,n))$ の値は，次式によって得られる．

$$\begin{bmatrix}4 & 2 & 0 & & \mathbf{0}\\ L_2 & 2(L_1+L_2) & L_1 & & \\ & \cdot & \cdot & \cdot & \\ & \cdot & \cdot & \cdot & \\ & L_{n-1} & 2(L_{n-2}+L_{n-1}) & L_{n-2} & \\ \mathbf{0} & 0 & 2 & 4 & \end{bmatrix}\begin{bmatrix}P_1'\\ P_2'\\ \cdot\\ \cdot\\ P_{n-1}'\\ P_n'\end{bmatrix}$$
$$=\begin{bmatrix}6D_1/L_1\\ 3(L_1^2D_2+L_2^2D_1)/L_1L_2\\ \cdot\\ \cdot\\ 3(L_{n-2}^2D_{n-1}+L_{n-1}^2D_{n-2})/L_{n-2}L_{n-1}\\ 6D_{n-1}/L_{n-1}\end{bmatrix} \qquad (3)$$

$$D_i=p_{i+1}-p_i,\ (i=1,2,\cdots,n-1).$$

(3)式の係数行列は3重対角であるから，この連立方程式は容易に解ける．なお，$P_i'(0)=P_i'$, $P_i'(L_i)=P_{i+1}'$ $(i=1,2,\cdots,n-1)$ の関係がある．

3. 3次スプライン関数に対する最適データ点

一般に，区分的補間法はデータ点数を増加するほど折線による補間に近付く．しかし，補間処理を高速化するため，あるいはデータを保存するためには，用いるデータ数は少ないほど望ましい．もし，一つの形状を表現するのに十分な必要最小限のデータ数が存在するものとすれば，この最小数の多寡はデータ点の配置に依存する．すなわち，形状変化の激しい部分では密に，逆に変化の緩慢な部分では疎に分布する，必要データ数が最小という意味で最適なデータ点の配列が存在すると考えられる．また，ある一定数のデータ点

列を用いる場合，これらのデータの配置が適切であれば，原曲線のもつ形状に関する情報を十分に引き出すことができる．したがって，本論文で研究の対象となっている3次スプライン補間を行う場合には，どのようなデータ点の配置が適切なものであるかを知る必要がある．

適切なデータ点列であるか否かを調べるためには，補間曲線の良さの定量的な議論が必要である．そこで，次の補間誤差に関する定義を行う．

[補間誤差]

既知の一価関数 $g(t)$ $(a\leqq t\leqq b)$ から得られる n 個のデータが，ある補間関数 $Jg(t)$ によって補間されているものとする．このときの補間誤差を次式で定義する．

$$\|g(t)-Jg(t)\|e_n=\max_{a\leqq t\leqq b}|g(t)-Jg(t)| \qquad (4)$$

この補間誤差の定義は，補間曲線にしばしば現れるうねりを調べるのに都合が良い．例として，古くから研究が行われている Lagrange 補間関数（例えば文献6)）を挙げることができる．Lagrange 補間関数を用いて補間を行った場合，データ点の与え方によってはルンゲの現象[7]として知られている激しく振動した補間曲線が得られることがある．この振動は，$\|\|e_n$ の増加として定量的に取り扱うことができる．また，$\|\|e_n$ は折線近似の近似誤差の評価に次のように用いることができる．

[定理1][5]

$g(t)\in C^2(a,b)$ であり，$|g''|$ は a および b 付近で単調で，$\int_a^b|g''(t)|^{1/2}\mathrm{d}t<\infty$ とすると，

$$|g''(t)|^{1/2}\mathrm{d}t=\frac{i-1}{n-1}\int_a^b|g''(t)|^{1/2}\mathrm{d}t \qquad (5)$$

$$i=2,3,\cdots,n-1$$

によって与えられる点列 $t_1(=a),t_2,t_3,\cdots,t_n(=b)$ を用いて a，b 間を補間することにより，

$$\|g(t)-J_2g(t)\|e_n=O(n^{-2})$$

となる．ここで，J_2 は折線補間を表す．

定理1において O はオーダを表している．この定理1によって，(5)式に従ってデータ点を選択し，折線によって関数の補間を行った場合の補間誤差は，データ数の2乗の逆数のオーダで減少することが保証される．

それでは，本論文で目標としている3次スプライン補間に適したデータ点列は，どのようにすれば得られるであろうか．いま，被補間関数 $g(t)$ が十分滑らか（高階導関数まで連続）であると仮定する．このような $g(t)$ を3次スプライン関数を用いて近似する場合，$g(t)$ 上のデータ点の位置を，どのように選んだ場合に近似誤差が小さくなるであろうか．この問題を取り扱うため，次の定義をする．

[2階導関数補間誤差]

既知の一価関数 $g(t)$ $(a\leqq t\leqq b)$ から得られる n 個のデータが，ある補間関数 $Jg(t)$ によって補間されているものとする．このときの2階導関数補間誤差を次式で定義する．

$$\|g(t)-Jg(t)\|''e_n=\max_{a\leqq t\leqq b}|g''(t)-(Jg)''(t)| \qquad (6)$$

ところで，補間関数の目標である視覚的に自然な曲線についての明確な数学的定義は現在までのところ与えられていない．また，補間曲線の視覚的自然さとその補間曲線の導関数がもつべき連続性との関係も議論の余地があるところであるが，CAGD における自由曲面創成においては C^2 級（2階導関数まで連続な関数の集合）が一つの基準になっている[10]．これは，2階導関数の連続性が曲率半径の連続性に深く関係しているためと考えられる．上の定義は，このことを考慮して導入した．2階導関数補間誤差の定義と定理1により次のことがいえる．

[定理2]

$f(t)\in C^4(a,b)$ について $f(t)$ の2階導関数が定理1の $g(t)$ に関する条件を満足すれば，

$$\|f(t)-Cf(t)\|''e_n=O(n^{-2})$$

となる．ここで，$Cf(t)$ はデータ点上での2階導関数が $f(t)$ のそれに一致するような3次スプライン補間関数を示す．

この定理は，データ点上で2階導関数が一致するような3次スプライン関数は，原曲線の2階導関数を $O(n^{-2})$ の精度で近似することを示している．さらに，$Cf(t)$ について，次のことがいえる．なお，定理2においては定理1の $g(t)$ との混同を避けるため $f(t)$ を用いたが，以下の記述において，関数名を $f(t)$ から再び，$g(t)$ に戻す．

[定理3]

$g(t)\in C^4(a,b)$ について $g'(a)=(Cg)'(a)$ であれば，

$$\|g(t)-Cg(t)\|e_n=O(n^{-2})$$

となる．

(証明)

$Cg(t)$ の定義により，

$$g''(t_1)=(Cg)''(t_1),\ g''(t_2)=(Cg)''(t_2),\cdots,$$
$$g''(t_n)=(Cg)''(t_n),$$

となる．いま，

$h(t)=g(t)-Cg(t)$

とすれば，

$h(t)=h(a)+th'(a)+t^2h''(\xi)/2,$

$a\leqq\xi\leqq b$

であるから，

$\|h\|_{e_n}\leqq|h(a)|+(b-a)|h'(a)|$

$+(b-a)^2\|h\|''_{e_n}/2$

$h(t)$ の定義より，

$h(a)=0$

また，条件より，

$h'(a)=0$

であるから，上のことがいえる．

(証明終)

なお，$C^4(a, b)$ クラスの関数を考えているので，4階導関数を用いて適切にデータ点を選択すれば（文献5)，p. 189 参照)，$O(n^{-4})$ の収束が得られる．ここでは2階導関数を用いているため，収束は $O(n^{-2})$ であるが，4章の例に示されるように，実用上，十分良好なデータ点列選択法が得られる．

条件 $g'(a)=(Cg)'(a)$ は，2章で述べた強制スプラインを与える．しかしながら，計算されるスプライン関数の両端点における条件の差異による変化は，局所的であると見なすことができる．したがって，両端点の外側に数個の補助的なデータ点を導入すれば，条件 $g'(a)=(Cg)'(a)$ を考慮することなく，高精度で，定理3を適用することができる．そこで，この定理の結果を明示するために，次の定義を与える．

[最適データ点列]

一価関数 $g(t)$ $(a\leqq t\leqq b)$ から n 個のデータ点を(5)式に従って選び $(t_1^*, p_1^*), (t_2^*, p_2^*), \cdots, (t_n^*, p_n^*)$ とする．この点列を $g(t)$ に対する最適データ点列という．

なお，前述のように，$C^4(a, b)$ クラスの関数を考えているので，$O(n^{-4})$ の収束をする点列を得ることが可能で，本来なら，これを最適データ点列と呼ぶべきである．しかし，本論文では2階導関数を中心にして議論しているので，定理3より得られる点列を最適データ点列と名付けた．

最適データ点列が実際に視覚的に自然な補間曲線を与えるのに役立つかどうかを検証するための具体例を次章に示す．

4. 最適データ点列の応用

[最適データ点列]は，条件 $f''(t_i)=(Cf)''(t_i)$ (すなわち $g'(t_i)=(Cg)'(t_i)$) が高精度で満足されるとき，真に最適なデータ点列を与える．一方，データ点数を増加させてゆけば，この条件が精度良く満足されるようになる．したがって，[最適データ点列]を用いるということは，原曲線と補間関数が一致するための2階導関数を用いて表現された条件によって，同じデータ点数であれば，極力，原曲線に忠実な補間関数が得られるように，データの位置を選択するための手法を与えることになる．

(A) 最適データ点列の生成

原関数 $g(t)$ が既知である場合は(5)式を用いてデータ列を生成すれば3次スプライン補間のための最適データ点列が得られ，データ数の減少化を図ることができる．例として，関数 $g(t)=1/(1+25t^2)$ を考える．この関数は前述のルンゲの現象を生ずることが知られている．図2にデータ数が8個の場合の(i)等間隔データに対する3次スプライン補間関数，および(ii)最適点列配置に対する3次スプライン補間関数を示す．

この関数においては，データ数が偶数の場合，関数値最大点 $(t=0)$ にデータが存在しないため，図2のようなうねりを生ずる（第1，第2，第6および第7セグメント)．しかし，(ii)においては，$|g''(t)|$ が大きくなる部分でデータの分布が密になるため，うねりがかなり減少しているのが理解できる．

原関数が陽に与えられておらず，点列の集合として与えられている場合は，点列 $((t_i, p_i); i=1, 2, \cdots, n)$ から次式によって定義される差分商を計算する．

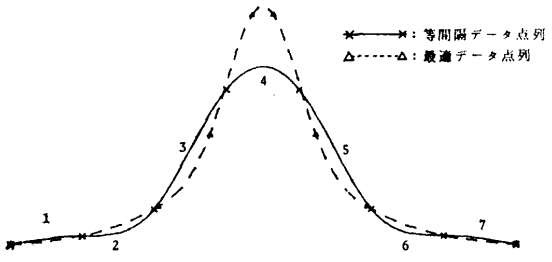

図2 ルンゲの例に対する最適データ点列
データ点は8点で実線が等間隔データに，破線が最適データ点列に対する3次スプライン補間関数．数字は等間隔データのセグメント番号．

Fig. 2 Proper data sequence for Runge example. Eight data points of equal spacing are interpolated by the cubic spline (solid line), and the same number of the data of proper spacing are also interpolated by the cubic spline (dashed line). Numbers indicate segment numbers of equal spacing data.

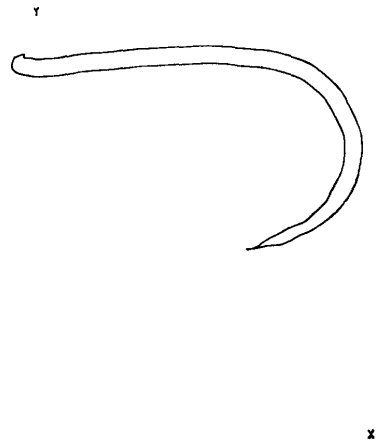

図 3 手書き文字のスプライン補間
36個のデータを1組のスプライン曲線で補間した例.

Fig. 3 Cubic spline interpolation for a handwriting letter data.
Thirty six data were interpolated with a pair of cubic spline interpolants.

$$\Delta p_i=(p_{i+1}-p_i)/(t_{i+1}-t_i),\quad i=1,2,\cdots,n-1, \tag{7}$$
$$\Delta^2 p_i=(\Delta p_{i+1}-\Delta p_i)/(t_{i+2}-t_i),\quad i=1,2,\cdots,n-2.$$

次に，二階差分商 $\Delta^2 p_i$ を(5)式の $g''(t)$ の代わりに用いて，積分を台形積分などの数値積分で置き換えれば，最適データ点列を計算できる.

図3に手書き文字入力装置により得られた36点のデータを一組のスプライン関数を用いて補間した例を示す[11]. 各データ点はそれぞれ X 座標，Y 座標および筆圧に関するデータを有しており，一組のスプライン関数は，X, Y 座標によって定義される位置データを筆圧データに従って肉付けする際の一組の点列をそれぞれ補間したものである[11]. X 座標，Y 座標および筆圧データについて独立に(7)式で与えられる差分商を計算し，さらに(5)式で表される積分値を台形積分法によって計算する. このときの被積分関数を次式によって定める.

$$\Delta^2 p_i=w_x\Delta^2 p^x{}_i+w_y\Delta^2 p^y{}_i+w_h\Delta^2 p^h{}_i \tag{8}$$

ここで，$\Delta^2 p^x{}_i$, $\Delta^2 p^y{}_i$, $\Delta^2 p^h{}_i$ は(7)式によって得られる各成分の二階差分商であり，w_x, w_y, w_h は各成分の重み係数である. これらの重み係数は，それぞれのデータのデータ点選択に寄与する割合を，応用の目的に応じて変化させるために導入した. 与えられた36個のデータ点列からほぼ1/2の17個の点を $w_x=w_y=1.0$, $w_h=0$ として提案手法により選択し一組のスプライン関数を計算したものを**図4**(a)(実線)に示す. この図から分かるように，この例では提案手法により，データ数をほぼ1/2に減少させても，原型をほとんど損なわない手書き文字が得られることが理解できる. 図4(b)(実線)にデータ数をさらに原データの1/4の9点に減少させたものへの適用例 ($w_x=w_y$

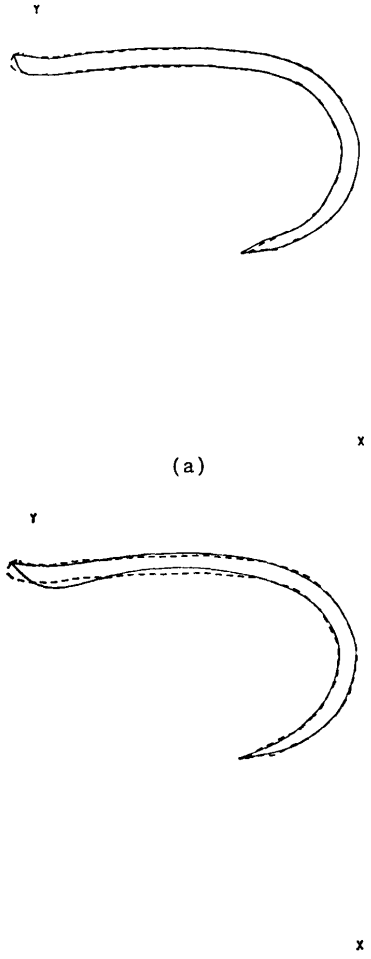

図 4 提案手法による手書き文字のスプライン補間
$w_x=w_y=1.0$, $w_h=0$ として提案手法により得られた(a) 17個および(b) 9個のデータを1組のスプライン曲線で補間した例. 実線が得られた補間曲線で点線は図3の原曲線を示す.

Fig. 4 Application of the proposal to the cubic spline interpolation for a handwriting letter data.
The cubic spline interpolation was carried out for (a) 17 and (b) 9 data, obtained by the proposal with the parameter value of $w_x=w_y=1.0$, $w_h=0$. The solid line shows the obtained interpolant, while the dashed line indicates the original interpolant of Fig. 3.

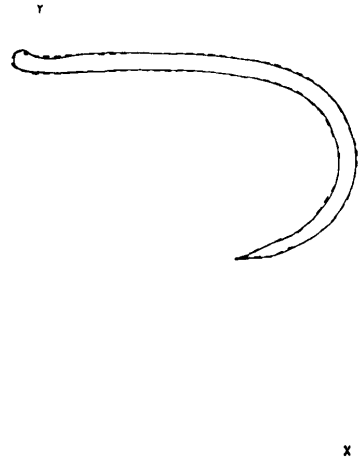

図 5 提案手法による手書き文字のスプライン補間
$w_x=w_y=1.0$, $w_h=0.2$ として提案手法により得られた 17 個のデータを1組のスプライン曲線で補間した例. 実線が得られた補間曲線で点線は図3の原曲線を示す.

Fig. 5 Application of the proposal to the cubic spline interpolation for a handwriting letter data.
The cubic spline interpolation was carried out for 17 data obtained by the proposal with the parameter value $w_x=w_y=1.0$, $w_h=0.2$. The solid line shows the obtained interpolant, while the dashed line indicates the original interpolant of Fig. 3.

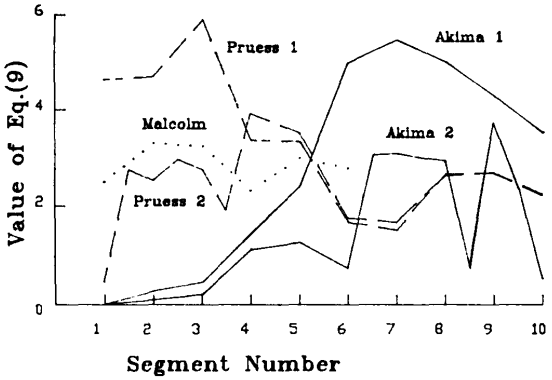

図 6 三つの例に対する(9)式の値
Malcolm, Akima 1, Pruess 1 はデータ点追加処理を適用する前の(9)式の値であり, Akima 2, Pruess 2 はこの処理を適用した後の値の各セグメントにおける変化を示す.

Fig. 6 Value of Eq. (9) in the text for the three examples.
Malcolm, Akima 1 and Pruess 1 indicate the variation of Eq. (9) before the new data insertion algorithm is applied, while Akima 2 and Pruess 2 show the same value variation along the segment after the algorithm is applied.

$=1.0$, $w_h=0$) を示す. データ数がこの程度まで減少すると,「つ」の字の湾曲部は提案手法により忠実に表現されるが, 直線に近い部分はデータ不足のためうねりを生じているのが観測される. 次に, (8)式における重みの影響を調べるため, $w_x=w_y=1.0$, $w_h=0.2$ として提案手法を適用する. **図5** (実線) に図4(a)の場合と同数の 17 点をこの重みを用いて得たものに対し一組のスプライン関数を計算した結果を示す. 図5と図4(a)とを比較すると, (8)式から予想されるように, 前者において, 筆圧変化の激しい文字の書き始めおよび書き終わりの部分が忠実に表現されているのが理解できる.

このように, 提案手法は, データの各特徴量に対し, 適切な重みを設定して適用すれば, 問題に即した最適解を求めることが可能となる.

(B) データ点列の評価

データ点列が既に与えられている場合 (一般に実験によって得られたデータはこの場合に相当する), この点列が, 3次スプライン補間を適用した場合に十分であるかどうか評価する. この目的のため, (5)式の右辺の値を得られた3次スプライン関数について計算する. つまり, 与えられたデータ点列に対し, 各セグメントごとに次の計算を行う.

$$\int_0^{L_i} |6\alpha_i u+2\beta_i|\,du \qquad (9)$$

ただし, α_i および β_i は, それぞれセグメント i における(2)式の u^3 および u^2 の係数である. (9)式の値を, Malcolm[12], Akima[13] および Pruess[14] の論文中の例に適用した結果を**図 6** (Malcolm, Akima 1, Pruess 1) に示す. また, 各例に対する3次スプライン補間を**図 7**, **図 8** (実線) および**図 9** (実線) に示す. これらの結果から分かるように, Malcolm の例は, データ点が適切に選んであり, うねりは生じていない. これに対し, 他の2例では, (9)式の値の変化が大きいことが分かる. したがって, 例えば Akima の例の第 6-9 セグメント, および Pruess の例の 1-3 セグメントにおいて, データが不足していることが分かる. 可能であれば, 実験もしくは観測を再度行い, この部分にデータを追加すべきであることが理解できる.

(C) データ追加法への適用

データが不足していることが明白であるが, 追加すべきデータを得ることができない場合を考える. この問題は, 不足したデータを推定し, これを自動的に補うもので, 次善の策を与えるものである. このための

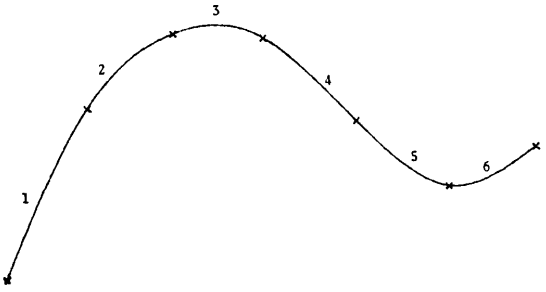

図 7 Malcolm の例への3次スプライン補間関数
Fig. 7 Cubic spline interpolant for Malcolm's data.

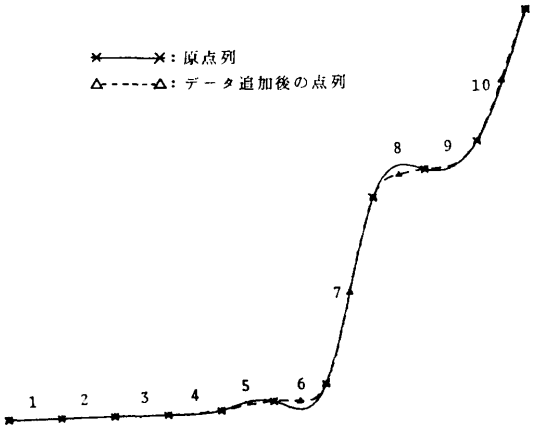

図 8 Akima の例への3次スプライン補間関数
実線は原点列，破線はデータ追加法（4点追加）を適用した点列への3次スプライン補間関数．数字は原点列のセグメント番号．
Fig. 8 Cubic spline interpolant for Akima's data. Solid line shows the interpolant for Akima's data, and dashed line is the interpolant for the whole data (four new data have been generated) by the proposed method. Numbers indicate segment numbers of original data.

データ追加法としては，その目的に応じて種々のものが考えられる．ここでは目的をスプライン関数に生ずるうねり除去という点にしぼる．3次スプライン曲線におけるうねりは，スプラインに張力を導入することによって取り除くことができる[15]．しかし，この方法を用いると，入力パラメータ数が増加し，本論文で考えている補間の目的である計算機による自動処理という見地からは不都合である．そこで，うねり発生によって(9)式で与えられる積分値が増加することに着目し，このようなセグメントにデータ点を追加することによりうねりを除去することを考える．

ところで，3次スプラインは，Bスプライン等と異なり，局所性を有しない（例えば文献 9)）．したがって，一つのデータの追加は，曲線全体に影響を与える．しかし，変更の生じたデータから3点程度離れれば，この影響による曲線形状の変化は，ほとんど無視できる[9]．そこで，新データ (t_m, p_m) を互いに隣接するデータ (t_i, p_i) および (t_{i+1}, p_{i+1}) によって定義されるセグメントの中間に次式に従って挿入する．

$$t_m=(t_i+t_{i+1})/2,$$
$$p_m=\{P_i(u_m)+(p_i+p_{i+1})/2\}/2,$$
$$u_m=t_m-t_i. \qquad (10)$$

なお，データを追加すべきセグメントとして，(9)式で与えられる積分値が最大のものが選ばれる．データ点追加によりうねり除去法として "Taut" 3次スプライン補間を用いるもの[5]があるが，この手法ではうねりを，余分な変曲点の存在として捉えている点が，本論文の手法と異なる．

本アルゴリズムの基本的な考え方は，データが相対的に不足しているセグメントの中間に，この点の独立変数の値に対応する(i)スプライン曲線上の点 $(p_i(u_m))$ と，(ii) (t_i, p_i) および (t_{i+1}, p_{i+1}) によって定義される直線上の点 $((p_i+p_{i+1})/2)$ の平均値として新しい従属変数の値を定めることにある．この追加点は，不足しているデータの位置を高精度で推定することにより得たものではないが，データが不足していることによって生ずるうねりを小さくする効果があ

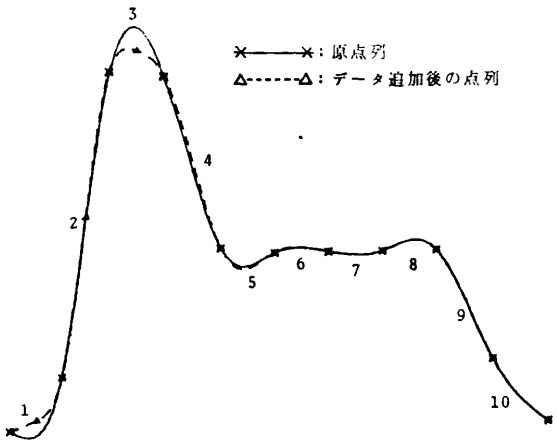

図 9 Pruess の例への3次スプライン補間関数
実線は原点列，破線はデータ追加法（3点追加）を適用した点列への3次スプライン補間関数．数字は原点列のセグメント番号．
Fig. 9 Cubic spline interpolant for Pruess' data. Solid line shows the interpolant for Pruess' data, and dashed line is the interpolant for the whole data (three new data have been generated) by the proposed method. Numbers indicate segment numbers of original data.

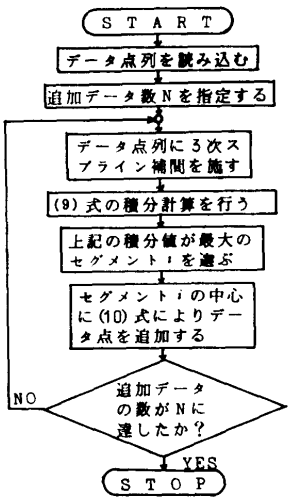

図 10 データ点追加アルゴリズム
Fig. 10 Algorithm for data insertion.

る．また，追加データ点をセグメントの中間に選んでいるのは，提案手法では，各セグメントの(9)式によって与えられる積分値を監視しながらデータ点を繰り返し追加してゆくため，各セグメントの相対的な長さの差を小さく保つためである．

(C)の具体的方法を**図 10** のアルゴリズムに示す．このアルゴリズムを繰り返し適用し，データ点追加を行う．追加するデータ数は全データ数の半数以下とする．これは，追加するデータ数が多すぎると，原曲線の形状が追加されたデータに強く支配されてしまい，原データのもつ意味が薄れてしまうのを避けるためである．また，Malcolm の例は，もともとデータ点が適切に選んであり，うねりも生じていないので，ここでは，Akima および Pruess の例に (C) の手法を適用する．

Akima の例にデータを追加したものを図8（破線）に示す．図6に示してあるように，Akima の例においては(9)式の値が4以上になるセグメントが4個あるので，4個のデータを追加することにする．提案アルゴリズムにより，新データは第7，第6，第8および第 10 セグメントの順に挿入される．第9セグメントにデータが挿入されなかったのは，第8セグメントに挿入されたデータのために，第9セグメントにおける（9）の積分値が減少し，第 10 セグメントのそれより小さくなってしまったためである．図8（破線）から分かるように，第6，第8および第9セグメントのうねりが，ほとんど消失している．

図9（破線）は Pruess の例への適用である．この例では，図6を参考にして，(9)式で与えられる積分値が4以上のセグメント数である3個のデータを追加した．追加データは，第3，第2および第1セグメントの順に挿入された．この挿入により，第1セグメントのうねりが小さくなっている．なお，第3セグメントにおいては，関数値最大点のデータが欠如しているので，このセグメントにおいてうねりが発生しているか否かの判断は困難である．ここではうねりとして処理したが，図 10 のアルゴリズムに，データ点追加禁止セグメントを指定する機能を付加すれば，第3セグメントを処理しないようにできる．

図6 (Akima 2, Pruess 2) に本節のデータ点追加手法を適用後の各セグメントにおける(9)式の値の変化を示す．これらの折線から分かるように，提案手法により，極端に(9)式の値が大きくなる場所，換言すれば，極端にデータの情報が不足している場所が消失している．

5. あ と が き

3次スプライン曲線による補間に適した点列を，2階導関数に関する近似の良さという観点から定量的に解析した．そして，原関数が既知である場合，もしくは関数の値が点列の集合として与えられている場合に最適データ点列を得る方法，与えられたデータ点列の良否を定量的に判定する方法，および，うねり除去を目的とした簡便なデータ追加アルゴリズムを与えた．

提案手法は，(1)式で示される一価関数に対し，最適データ点列を求めることにより得られたものであるが，手書き文字の処理の部分で用いた簡便な方法のように多価関数に拡張することができる．この簡単な方法は補間問題でしばしば用いられるパラメトリック手法[6]を適用して得られた．しかしながら，多価関数を厳密に扱う場合，X 成分，Y 成分等の各成分間の相互関係が問題となるならば，これらをも考慮して最適データ点列を定めなければならない．さらに，提案手法を用いてリアルタイムで最適データ点列を発生する場合は，(5)式の a, b を定めることができず，この積分を過去のデータの平均値として動的に変更しなければならない．このための具体的な手法も残された課題である．

参 考 文 献

1) Foley, J. D. and Van Dam, A.: *Fundamentals of Interactive Computer Graphics*, Addison-Wesley (1982).
2) Faux, I. D. and Platt, M. J.: *Computational*

Geometry for Design and Manufacture, Ellis Horwood (1978).
3) Brodlie, K. W.: A Review of Methods for Curve and Function Drawing, in *Mathematical Methods in Computer Graphics and Design* (Brodlie, K. W. ed.), Academic Press (1980).
4) 佐藤：平面曲線の最適近似における適切な選択点数の決定法，電子通信学会論文誌，Vol. J 68-D, No. 2, pp. 199-200 (1985).
5) de Boor, C.: *A Practical Guide to Splines*, Springer-Verlag (1978).
6) Rogers, D. F. and Adams, J. A.: *Mathematical Elements for Computer Graphics*, McGraw-Hill (1976).
7) 市田，吉本：スプライン関数とその応用，教育出版 (1979).
8) Vandergraft, J. S.: *Introduction to Numerical Computations*, Academic Press (1983).
9) 原田，中前：閉曲線点列の3次スプライン補間問題における「うねり」除去の一手法，情報処理学会論文誌，Vol. 23, No. 3, pp. 219-225 (1982).
10) Whelan, T.: A Representation of a C^2 Interpolant over Triangles, *Comput. Aided Geom. Des.*, Vol. 3, pp. 53-86 (1986).
11) 原田，佐伯，中前：手書き文字作成のための筆圧入力装置の開発，電子情報通信学会技術研究報告，IE 87-86, pp. 9-15 (1987).
12) Malcolm, M. A.: On the Computation of Nonlinear Spline Functions, *SIAM J. Num. Anal.*, Vol. 14, No. 2, pp. 254-282 (1977).
13) Akima, H.: A New Method of Interpolation and Smooth Curve Fitting Based on Local Procedure, *J. ACM*, Vol. 17, pp. 589-602 (1970).
14) Pruess, S.: An Algorithm for Computing Smoothing Splines under Tension, *Computing*, Vol. 19, pp. 365-373 (1978).
15) Nielson, G. M.: Rectangular Splines, *IEEE Comput. Graph. Appl.*, Vol. 6, No. 2, pp. 35-40 (1986).

(平成元年3月22日受付)
(平成元年9月12日採録)

原田 耕一 (正会員)

昭和25年生．昭和53年東京工業大学大学院理工学研究科博士課程修了．同年高松工業高等専門学校電気工学科勤務．昭和54年広島大学工学部第2類（電気系）助手．昭和60年同講師．平成元年4月広島大学総合科学部助教授，現在に至る．この間，昭和57年12月より1年8か月間，米国クラークソン大学に客員研究員として滞在．また昭和63年1月より3か月間，バンコク市のアジア工科大学を訪問し，コンピュータ・グラフィックスに関する講義を行う．現在，コンピュータ・グラフィックスを中心としたデータ処理技術の研究に従事．工学博士．電子情報通信学会，ACM 各会員．