はん用 CAD における B スプライン曲線の 局所修正法に関する研究*

本田 知己^{*1}, 谷口 直之^{*2}, 岩井 善郎^{*1}

Local Modification Method for B-Spline Curve in the General CAD System

Tomomi HONDA, Naoyuki TANIGUCHI and Yoshiro IWAI

Recently, computer-aided design (CAD) systems are being used for total design from conceptual design to detailed design. Designers desire a CAD system which can draw their image faithfully. However, it is considerably difficult to modify the drawn curve in the present popular CAD systems. In this study, we developed a new local modification method for a B-spline curve by performing an inverse transformation of the B-spline curve. The influential range of each control point becomes narrow because of the increase of the number of control points, so that it is possible to modify the curve locally with mathematical smoothness. Furthermore, our developed method was linked to the general CAD and its availability was confirmed from the standpoint of execution time and handling.

Key Words: Design, Design Engineering, CAD, Curve Generation, Modification Method, Spline Curve

1. 緒 言

概念設計から詳細設計までの一連の作業をコンピュータの支援によって行う設計の CAD 化が進んでいる。しかし設計の初期段階では、設計者(デザイナー)は CAD とあまりうまくコミュニケーションできていないようである。これは CAD が直線や円といった幾何学的に特定な形状は正確に表現できるが、デザイナーがイメージすることの多い自由な曲線の表現は不得意であることや、CAD による自由曲線の生成法は難解でその操作性もよくないことにも原因があるものと考えられる。そこでデザイナーがイメージした曲線を忠実に表現することが可能な CAD の手法の開発が研究されている。

デザイナーは形状をイメージしながら視覚化する際、まずイメージした形状の概略を描き、その後広い範囲にわたる大まかな修正を行い、徐々に範囲を狭めながら細かな修正を行う。そのためデザイナーの視点グローバルなものからローカルなものへと移るのに伴って自由曲線を局所的に修正することが必要にな

る。

自由曲線の修正法として、Bezier 曲線を用いた方法が報告されている⁽¹⁾。これはマウスを用いてイメージに近い修正を容易に行うことができるが、修正箇所を数学的に滑らかにつなぐことが困難とされている。

本研究では、自由曲線として数学的に滑らかであり、かつ複雑な形状を描く場合にも与える制御点数が少ないスプライン曲線について、はじめとは異なった制御点の組合せに逆変換することにより、局所的に自由に修正できる手法を開発し、その有効性を考察した。

2. スプライン逆変換の アルゴリズムと計算

本研究で用いた三次 B スプライン曲線は、4 個の制御点によってのみ決定されるので、各制御点の影響は曲線全体に及ばない。そのため制御点の位置を変更した場合でも Bezier 曲線のように曲線全体が変形することはなく、曲線を局所的に修正することが可能である。また曲線は C^2 級の連続性をもつ曲線で滑らかである。しかし、制御点は曲線決定のための指標でしかないので、曲線は制御点を通らずに生成されるので、与えた制御点からその曲線形状を予測することは困難である⁽²⁾。

* 原稿受付 1994 年 12 月 19 日。

^{*1} 正員, 福井大学工学部 (〒910 福井市文京 3-9-1)。

^{*2} 学生員, 福井大学大学院。

このような特徴をもつ三次 B スプライン曲線を局所的に修正する場合を図 1 を例にして考えてみる。図 1 に示すように、デザイナーの視点が領域 I のようなグローバルなものにあるとき、中央のくぼみの形状を変えるために③の制御点を③'に移動した際に形状が変化する範囲はデザイナーの視点とほぼ同じである。しかし、デザイナーの視点がローカルな領域 II にあって中央のくぼみだけを局所的に変えようとしたときには、制御点③を③'に移動したのでは、変形の範囲が視点よりも広くなり曲線の形状を変化させたくない部分まで変形させてしまう。このようなときには、図 2 に示すようにさらに制御点②→②'、④→④'に移動して余計な部分に及んだ変形を試行錯誤を繰り返して元に戻す操作が必要となり、デザイナーにとっては大きな負

図 1 デザイナーの視点の変化とスプライン曲線の修正

図 2 デザイナーの視点がローカルな場合の従来の方法によるスプライン曲線修正法（図中の制御点は図 1 に対応する）

担になる。このようなことを避けるためには、視点の変化にあわせて個々の制御点が曲線に与える影響の範囲を変える必要がある。本研究では、スプライン曲線を逆変換し、制御点を増加させることにより自由曲線の局所的な修正を可能にしようとした。以下その方法について述べる。

CAD におけるスプライン曲線生成の計算の概略を図 3 に示す。通常は制御点群 A を与えてデータ点群 C を求めるのに対して、逆変換とはデータ点群 C より制御点群 A とは異なる制御点群 B を計算することである。

三次 B スプライン曲線は三次曲線の集合であるが、CAD ではそれらを折れ線によって近似して、折れ線のデータ点の数は

$$(\text{制御点間の分割区分数}) \times (\text{制御点数} - 1) + 1$$

で決まる⁽³⁾。

したがって、図 3 のようにそれぞれ異なる制御点群 A と制御点群 B から同じデータ点群 C を生成するものと仮定すると、データ点の数から以下の関係式が成り立つことになる。

$$\begin{aligned} & (\text{kubun}) \times (n - 1) + 1 \\ & = (\text{new} - \text{kubun}) \times (m - 1) + 1 \dots\dots\dots (1) \end{aligned}$$

{

kubun = 制御点間の分割区分数

new - kubun = 新たな制御点間の分割区分数

n = 制御点数

m = 新たな制御点数

}

これを m について解くと次式ようになる。

$$m = \left(\frac{\text{kubun}}{\text{new} - \text{kubun}} \right) \times (n - 1) + 1 \dots\dots\dots (2)$$

図 3 スプライン曲線生成の計算

ここで m は制御点群 B の制御点数であるため、式 (2) において $kubun/new - kubun$ は自然数であることが必要である。したがって、

$$\frac{kubun}{new - kubun} = \text{自然数} \quad \dots\dots\dots (3)$$

が成り立つとき、異なる制御点群 A, B より同じデータ点群を生成することが可能になる。これはいい換えると、式 (3) が満たされる時制御点群 A から生成したスプライン曲線を制御点群 B に逆変換できることを示す。

式 (3) で自然数の値は任意に設定することができるので、一度の逆変換で制御点数を極端に増加させたいときには大きな値を、逆に制御点数をあまり増加させたくない場合には小さな値をとればよいことになる。本研究では自然数の値は 2 とする。これは必要以上に制御点が増加することを避けるためである。また仮に逆変換後制御点が少なく曲線の修正に不都合が生じた場合でも、繰返し逆変換を行うことで必要な数だけ制御点を増加できるためである。

したがって、

$$m = 2n - 1 \quad \dots\dots\dots (4)$$

$$new - kubun = \frac{kubun}{2} \quad \dots\dots\dots (5)$$

となる。

次に、逆変換後の制御点の座標算出法について述べる。スプライン曲線のデータ点は次式によって計算される。

$$k_1 \times P[t](x, y) + \dots + k_4 \times P[t+3](x, y) = D(x, y) \quad \dots\dots\dots (6)$$

したがって、式 (6) をもとに逆変換後の制御点の座標 $RP[t](x, y), \dots, RP[t+3](x, y)$ を求める。ここで $k_1, \dots, k_4$ は三次 B スプラインであり、deboor-cox 法により決定される係数である⁽³⁾。これはスプライン曲線の次数と制御点数によってのみ決定され、データ点による影響はない。いま式 (4) より逆変換後の制御点数はわかっているので、逆変換後の三次 B スプラインは一義的に決定される。また $D(x, y)$ は図 3 における

データ点群 C の個々のデータ点で既知の値であるため、逆変換後の制御点 $RP[t](x, y), \dots, RP[t+3](x, y)$ を求めるためには未知数が 4 の連立一次方程式を解けばよいことになる。しかし、実際には曲線の始点と終点においてデータ点 $D[1](x, y), D[kubun \times (n-1)](x, y)$ と制御点 $RP[1](x, y), RP[m](x, y)$ はそれぞれ一致するため未知数は 3 となるので、未知数 3 の連立一次方程式を解けば逆変換後の最初の 4 個の制御点が求められる。次に、このようにして求めた 4 番目の制御点を基点にして、次の 3 個の制御点が計算できる。同様にして $(2n-1)$ 個の制御点が順次求められる。

本研究では、この連立一次方程式を解くために Gauss-Jordan 法を用い、計算時間を考慮して完全選択法ではなく部分選択法とした⁽⁴⁾。また曲線の終点 $RP[m](x, y)$ に近い制御点の計算では、計算誤差の累積やけた落ちなどの影響が生じると考えられる。そこで曲線の始点側、終点側の半分ずつに分けて、それぞれ始点と終点から曲線の中央部に向かって制御点を計算して、このような影響ができるだけ少なくなるようにした。

スプライン逆変換のプログラムは C 言語で作成し、はん用 CAD として広く利用されている Auto CAD for Windows R12J に組込んで実行した⁽⁵⁾。表 1 にシ

表 1 システム構成

C P U	PC-9821 Ap(i486DX2 66MHz)
R A M	640KB+14MB
HardDISK	240MB
O S	MS-DOS Ver5.0A-H MS-Windows Ver3.1
C A D	Auto CAD for Windows R12J

図 4 スプライン曲線の生成および修正手順

ステムの詳細を示す。またコンパイラには MS-C Ver 7 を用いた。

3. スプライン逆変換による 自由曲線修正法

スプライン曲線の生成および局所修正の手順の概略を図 4 に示す。スプライン曲線を生成する場合、制御点間の区分数をどの程度に選ぶかが重要になる。図 5 は、制御点間の区分数が異なるスプライン曲線を示したものである。区分数が増加するほどスプライン曲線は滑らかになり、より曲線らしくなることがわかる。このような変化を定量的に調べるために、区分数を変えて生成した種々のスプライン曲線の始点から終点までの長さを計算し比較した。

図 5 区分数の違いによるスプライン曲線の比較

図 6 は種々の区分数で得られるスプライン曲線の長さ非常に大きな区分数 (Segments=100) で生成した滑らかな曲線と考えられるスプライン曲線の長さとの比 (L) が区分数によって変化する様子である。図には曲線の生成に要する実行時間 (T) の変化もあわせて示す。滑らかなスプライン曲線との長さ比 L は区分数 3 で 0.995 4, 区分数 10 で 0.999 7, 区分数 20 で 0.999 9 となり、区分数の増加にともなって急速に 1.0 に近づく。また、実行時間は区分数に比例して増加する。したがって、図 5 の目視の結果および図 6 の計算の結果を総合すると、ここではスプライン曲線を生成する場合の区分数は少なくとも 10 以上で、20 程度が好適であると考えられる。そこで、逆変換を行う前の区分数を 20 とした。その結果、区分数 20 で n 個の制御点 (制御点群 A) を与えて生成したスプライン曲線は、区分数 10 で $m=2n-1$ 個の制御点 (制御点群 B) に逆変換できることになる。このようにして求めた逆変換後の制御点を用いて再びスプライン曲線を生成する。

図 7 は、スプライン曲線 [図 7 (a)] のデータ点をもとに 2 章で示した方法によって逆変換を行い、新たな制御点を求め曲線を描いた結果 [図 7 (b)] である。図 7 (b) は図 7 (a) と同じ形状になることがわかる。図 7 (c) はさらに制御点を増加した増合せである。制御点が増加するのに伴って制御点の位置は生成されるスプライン曲線に近づき、制御点から生成される曲線を予測しやすくなることがわかる。また図 7 (a), (b) の曲線について、それぞれ 1 個の制御点 (図中の ● 印で示す点) を下方に同じ距離だけ移動したとき (図中の ○ 印で示す点まで) の曲線の変化を図 8 に示す。修正曲線 (a) は元の曲線から広範囲にわたって変形している

図 6 種々の区分数で得られるスプライン曲線の長さ滑らかなスプライン曲線の長さ比 L および曲線生成時間 T の区分数による変化

図 7 スプライン逆変換による制御点の増加

のに対して、制御点を増加させて修正した修正曲線 (b) は元の曲線の中央部分だけが局所的に変形している。したがって、制御点を増加させると制御点が曲線の形状に関与する範囲は著しく狭くなることがわかる。

さらに、このような制御点の個数の影響を明らかにするために、隣合う制御点間を結んだ折れ線の累積長さ λ とスプライン曲線の長さとの比 (λ) をとって制御点数による変化を調べた。結果を図 9 に示す。図にはスプライン曲線逆変換に要した実行時間 (T) もあわせて示す。制御点数が増加すると λ は 1.0 に近づく。一方、実行時間は制御点数にほぼ比例して増加する。しかし、 $\lambda=1.01$ でそれ以降の変化も非常にゆるやかになる制御点数 17 の場合でも、実行時間は約 2 秒であった。

以上の結果から、スプライン曲線を逆変換し、制御

図 8 本方法によるスプライン曲線修正

図 9 (制御点を結んだ累積長さ)/(スプライン曲線長さ) λ および逆変換実行時間 T の制御点数による変化

点を増加することにより、個々の制御点の影響範囲を狭くすることができるので、例えば図 2 に示したように従来の方法で必要であった“曲線の局所修正における余計な部分に及んだ変形を元に戻す作業”は省けることが明らかになった。

4. 結 言

意匠設計において設計者(デザイナー)がはん用 CAD を用いてイメージした曲線を忠実に表現するための一支援手法として、スプライン曲線の局所修正法を研究した。その結果、スプライン曲線を逆変換し、曲線の形状を変えないように制御点を増加し個々の制御点の影響範囲を狭くすることにより、スプライン曲線を局所的に修正できることが明らかになった。

開発した手法のおもな特徴は以下のとおりである。

(1) 従来の方法では一つの制御点の影響範囲がデザイナーの視点よりも広い場合には、形状を変化させたくない部分にまで及んだ変形を元に戻す作業を強いられたが、本手法により局所的な修正ができる。

(2) 本手法は修正部分を点列で補間する方法など

と異なり、修正後も 1 本のスプライン曲線であるので連続性や滑らかさは失われない。

(3) 本研究で開発した手法をはん用 CAD に組み込み、CAD の機能の一部として利用できる。この場合逆変換に要する時間は短く、十分実用的である。

なお、今後この手法を発展させてスプライン曲線の修正が必要となる部分だけについて制御点を増加させる局所修正法を研究する予定である。

文 献

- (1) 福田・山崎, 機論, **60**-579, C(1994), 3657.
- (2) 山口, 形状処理工学 [I] [II], (1982), 日刊工業新聞社.
- (3) 吉村・高山, パソコンによるスプライン関数, (1988), 43, 東京電機大学出版局.
- (4) 内田, C 言語によるプログラミング(応用編), (1992), 282, オーム社.
- (5) オートデスク株式会社, Auto CAD ADS 入門, (1992), 2, オーム社.