

3次スプライン関数上の
サンプル点配置の一手法原田耕一・中前栄八郎
広島大学・工学部

コンピュータアニメーションにおいて広く用いられている3次スプライン関数上に、任意の規則で生成された相対距離をもつサンプル点列を配置する一手法について提案する。提案手法は大別して二つの部分から成る。すなわち、まずサンプル点列の各点の相対的位置を指定し、次に、得られる相対的位置関係に基づいてサンプル点列を3次スプライン関数上に実現する。これら二つのステップのアルゴリズムを与えるとともに、3次スプライン補間関数上で物体の動きをシミュレーションするためのサンプル点列を提示してその有用性を示す。

ON AN ALGORITHM FOR LOCATING
SAMPLE POINTS ON CUBIC SPLINE
INTERPOLANTS (in Japanese)

by Koichi HARADA and Eihachiro NAKAMAE

Faculty of Engineering, Hiroshima University
Saijo, Higashi-Hiroshima, Hiroshima, 724 JAPAN

This paper proposes a method to locate sample points on cubic spline interpolants, which have been widely used in the field of computer animation. The proposal consists of two parts: (i) to specify the relative position of the resulting sample point sequence depending on the prescribed rule, and (ii) to generate sample point sequence based on the relative position on the cubic spline interpolant. For demonstration, several examples are given which simulate movement of an object along the spline interpolant.

1. 緒言

3次スプライン補間^{[1], [2]}は、コンピュータ・グラフィックス技術を用いたコンピュータアニメーションおよび計算機援用形状設計問題 (CAGD) において広く用いられている。両者とも、最終的には人間によるインターアクティブな修正操作^{[3], [4]}が必要な場合が多いが、与えられたデータを基にした初期形状もしくは初期図形を創成するうえで3次スプライン関数は重要である。3次スプライン関数は与えられたデータ点を通過する多数の曲線セグメントの集まりであり、3次スプラインを計算するということはとりもなおさずこれらのセグメントを記述する係数を定めるといふことに他ならない。計算された3次スプラインのその後の処理は、セグメント上に設定される補間点 (サンプル点) によって行なわれる。

CAGDの分野においては一般に多数のサンプル点を各セグメント上に生成し、これらの点列を折線で結んで得られる曲線を補間曲線として代用している^[5]。したがって、隣接する個々のサンプル点間隔が均一であろうと不均一であろうと実用上差異はない。

一方、コンピュータアニメーションにおいては、パラメトリックキーフレーム法^[6]が広く用いられている。この手法においては、データ点に相当するのはキーフレームと呼ばれる複数のフレームの集合である。各キーフレーム間の画像を得るために、スプライン補間が用いられている。補間は各フレームに存在する物体単位に施され、隣接するキーフレーム間に補間フレームを生成するために、スプライン補間によって得られた物体データを基にして画像を構成する。二つの連続するフレームが提示される時間間隔は一定であるから、連続するフレームにおける各物体データの変化量、換言すればサンプル点列の変化量は物体の動きを反映したものでなければならない。この点で、コンピュータアニメーションとCAGDではサンプル点列の役割が本質的に異なる。

本論文では、コンピュータアニメーションにおけるような、補間関数上でのサンプル点列の相対的位置が問題となる場合を考え、予め規定されたサンプル点列の分布をいかにして3次スプライン補間関数上で実現するかについての理論的検討を行なう。すなわち、次の手順によって実現する手続きを提案する。

有限個のデータ点が与えられた場合 (1) これらのデータ点列を3次スプライン関数で補間する。(2) 物体の与えられた動作記述規則 (例えば等速度、等加速度など) に基づいて、補間関数上におけるサンプル点列の相対的位置を決定する。すなわち、最初および最後のサンプル点間の距離を補間

関数上におけるそれぞれのサンプル点に対応する位置の点の相互距離に等しくするよう線形変換を施すことによって実際のサンプル点相互間の距離を求める。(3) このようにして得られるサンプル点個々の相互関係を3次スプライン関数上で実現する。

本論文では、データは2次元平面上に与えられるものとし、補間曲線は平面曲線であるものとする。これは、適用例を表示するのに都合が良いので用いている仮定であり、パラメトリックに曲線を扱う場合には2次元と3次元とは本質的な差異はない^[1]。以下、スプライン関数の概要を述べ、上述の手順に従って数学的議論を展開する。さらに、提案手法の有効性を検証するために、アニメーションにおいて、製作者が希望するスムーズな動きを与える手法を例示する。

2. 3次スプラインの陰関数表現

2次元空間にn個のデータ点列 P_1, P_2, \dots, P_n が与えられたものと仮定する。周知のように、このデータ点列に対する3次スプライン補間曲線 (曲線セグメント) は次式によって与えられる^[1]。

$$x(t) = a_3^i t^3 + a_2^i t^2 + a_1^i t + a_0^i, \quad (1)$$

$$y(t) = b_3^i t^3 + b_2^i t^2 + b_1^i t + b_0^i,$$

$$0 \leq t \leq t_i, \quad i = 1, 2, \dots, n-1.$$

ここで i はスプラインセグメントの番号であり、 a_0^i, \dots, b_3^i は、第 i セグメント、すなわちデータ P_i および P_{i+1} の間で定義される係数である。スプライン関数を計算する際は、端条件と呼ばれる補間曲線の両端点における条件が必要である。ここでは、自然スプライン (両端点における曲率がゼロと仮定して得られる) を用いる。このスプラインを用いる理由は入力すべきデータを減少させるためと、端条件の補間曲線全体に及ぼす影響は一般に両端点付近に限られる^[7]との事実による。スプライン補間曲線を表示する際は式(1)で定義される曲線上に適当な個数のサンプル点を定める。この点列を折線で結んで得られる線分が実際の表示曲線となる。表示に必要なサンプル点の個数、およびパラメータ空間上でのサンプル点の位置の定め方は、単に曲線の表示を目的とする場合は重要ではなく、次の簡易法、[サンプル法1] が従来からよく用いられている。

[サンプル法1]

各セグメント上に m 個のサンプル点を生成するものとする。
 i 番目のセグメント ($i=1,2,\dots,n-1$) について、このセグメントのパラメータ空間上の長さ t_i を $m+1$ で割る。そして、 $t = t_{ij}/(m+1)$; $j=0,1,2,\dots,m$ によって得られるパラメータ値にしたがってサンプル点の座標を計算する。なお $j=m+1$ に対応する点は $i+1$ 番目のセグメントの第1サンプル点であり、かつデータ P_{i+1} に一致する。

この方法によれば、パラメータ空間では、連続するサンプル点の間隔は一つのセグメント内では均一となる。しかし、表示空間とパラメータ空間との写像は線形ではないので、表示空間では隣接するサンプル点の間隔は一般に不均一となる。

ところで、本論文では、表示空間とパラメータ空間との写像を解明し、希望するサンプル点列の間隔を表示空間で実現することが主目的である。これを実現するための表示空間とパラメータ空間との関係を調べるためには、Sederberg 等によって最近報告された^[9]陰関数化の手法が有用である。Sederberg 等の方法は、行列式の演算において広く知られている Sylvester の手法を出発点としている。彼らの方法を3次スプラインに適用するため、まず、式(1)を $a_3^i t^3 + a_2^i t^2 + a_1^i t + a_0^i - x = 0$ および $b_3^i t^3 + b_2^i t^2 + b_1^i t + b_0^i - y = 0$ の形に書き直す。すると Sylvester の手法により、式(1)の陰関数表現は次式で与えられる。

$$\begin{vmatrix} a_3^i & a_2^i & a_1^i & a_0^i - x & 0 & 0 \\ 0 & a_3^i & a_2^i & a_1^i & a_0^i - x & 0 \\ 0 & 0 & a_3^i & a_2^i & a_1^i & a_0^i - x \\ b_3^i & b_2^i & b_1^i & b_0^i - y & 0 & 0 \\ 0 & b_3^i & b_2^i & b_1^i & b_0^i - y & 0 \\ 0 & 0 & b_3^i & b_2^i & b_1^i & b_0^i - y \end{vmatrix} = 0 \quad (2)$$

式(2)は x および y に関して3次式となっている。記号処理ソフトウェアを使えばこの式は容易に展開できて、表1に示す個数の係数を持つ一つの方程式となる。なお、係数の個数の合計は102個である。

3. サンプル点の配置

本節において(2)式の表現を基にして表示空間とパラメータ空間との写像を定式化する。サンプル点配置問題は、次の二つのステップに別けて取り扱うことができる。第一段階は与えられた規則、例えば等速度あるいは等加速度といった

性質に基づいてサンプル点を規定することである。第二段階は、得られたサンプル点間の相対距離に応じてこれをスプライン曲線上に実現することである。第一、第二段階に分割して以下述べる。

3-1. サンプル点間の相互距離

議論の展開を容易にするために、サンプル点の生成は図1に示すグラフを用いて行なう。このグラフの横軸は P_1 を原点としたスプライン曲線に沿う長さによって定義される独立変数である。したがって、この独立変数の値が、スプライン曲線上の各データ点に対する相対位置を与えることになる。この軸を設定する場合、式(1)のパラメータ長 t_i ($i=1,2,\dots,n-1$) を用いてもよいが、サンプル法1により得られるサンプル点列によって定義される折線長を使用したほうがよりスプライン曲線の長さに近いものが得られる。

次に、このグラフの従属変数として曲率の近似値を用いる。もちろん、物体の動作記述の目的により他の量を設定してもよい。例えば物体が次第に大きくなるアニメーションにおいては物体の大きさが大きい程移動速度を下げることにより臨場感が得られることが知られている^[9]。この場合は、物体の大きさを独立変数とすればよいし、もし、物体の大きさが変化しながら曲線上を動く場合は、物体の大きさと曲線の曲率というように複数の、動きを決定する特長量を設定してもよい。ここでは単純に曲線の曲率のみを考慮して動きを決定する場合を考える。曲線 $y=f(x)$ の曲率は、厳密には、

$$\kappa = y'' / (1 + y'^2)^{3/2} \quad (3)$$

によって与えられる。ここで $'$ は x に関する微分を表わしている。スプライン(厳密には線形スプライン)曲線を代表とする補間曲線は $|y'| \ll 0$ の仮定のもとに導出されている。そこで、 y'' を近似的な曲率と考える。パラメトリックな表現 $x=x(t)$, $y=y(t)$ において、この近似的な曲率は、

$$y'' = (d^2y/dt^2 \cdot dx/dt - dy/dt \cdot d^2x/dt^2) / (dx/dt)^2 \quad (4)$$

となる。式(1)により各セグメント毎に $dx/dt, \dots, d^2y/dt^2$ を容易に求めることができる。しかし、 $|dx/dt|$ が小さい場合に式(4)の計算精度が問題となる。動きと曲率との関係を調べる場合、曲率の符号は重要ではなくその大きさが分かれば充分である。そこで、実用的な曲率表現 κ として次のものを考える。

$$K = \sqrt{(d^2x/dt^2)^2 + (d^2y/dt^2)^2} \quad (5)$$

この式は、x成分およびy成分のそれぞれのパラメータtに関する近似的な曲率が互いに直交し、これらによって決定されるベクトルの大きさが補間曲線上の対応する点の曲率の大きさを近似的に与えているものと仮定して導出したものである。

図1で与えられる曲率の変化をサンプル点列の配置にどのように反映するかの具体的な方法は応用により異なる。ここでは最も簡単な例として、式(5)で示されるKのスプライン曲線上に沿っての積分がサンプル点間で一定になるようにする。これは、曲率の大きい場所では低速で、曲率の小さい場所では高速で動く物体のシミュレーションと考えることができる。

このようなサンプル点列を得るには次の手順を踏めばよい。(a)式(5)のKをパラメータ空間で曲線全体(周期関数である場合は一周期分)に亘って数値積分する。(b)この積分値を、生成すべきサンプル点数で割り、隣接するサンプル点間での積分値kを算出する。(c)第一サンプル点から始めて、隣接するサンプル点間の数値積分がkとなるように、サンプル点列を図1の独立変数の空間において順次求める。

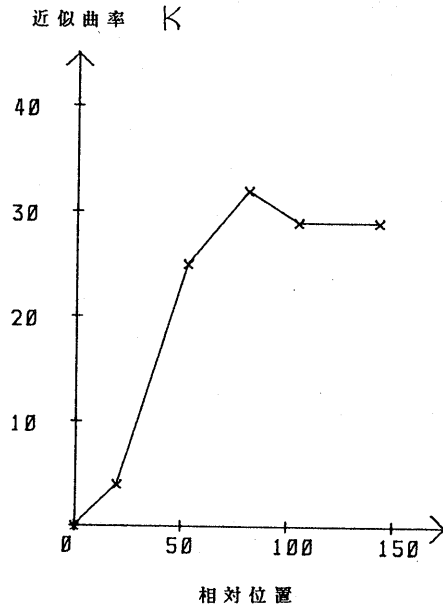


図1 スプライン上の相対位置と曲率

3-2. サンプル点列の実現

次の段階は得られたサンプル点間相互距離を補間曲線上において実現することである。補間曲線上に、このサンプル点列に対応する点列を計算する方法として次の2つのものが考えられる。

(a) 最初のサンプル点を最初のデータ点 P_1 に一致させ、2番目のサンプル点を式(2)で示される陰関数を解くことによって得る。2番目のサンプル点を得られれば、これを新たな初期値として第3のサンプル点を計算する。この手続きを最後のサンプル点を得られるまで繰り返す。

(b) 全体の流れは第1の方法と同じであるが、式(2)の陰関数を解く代わりに各サンプル点に対応するパラメータ値をパラメータ空間上に求める。

本研究では(b)の手法を用いる。なぜならば、(a)の方法においては、逐次計算の各ステップで双3次方程式を処理せねばならず、さらに対象とする補間曲線は一般に多価関数となり、収束計算における判定が複雑となるからである。

(b)の方法を用いるためには、式(2)の陰関数表現と式(1)のパラメトリック表現との間の写像を導出する必要がある。これには次の古典的定理が有用である。

「定理」

$F(x, y)$ が点 (p, q) を含む閉領域で連続微分可能であり、 $F(p, q) = 0$ の条件のもとで、もし、偏微分 $F_y(p, q) \neq 0$ でないならば、 $F(x, f(x)) = 0$ を満足する一価関数 $f(x)$ が存在し $q = f(p)$ となる。また、関数 $y = f(x)$ は微分可能であり、

$$dy/dx = -F_x(x, y) / F_y(x, y). \quad (6)$$

となる。

いま、 (p, q) および (p', q') を式(2)で与えられる曲線上の相異なる二つのサンプル点であるとする。これらの間隔は一般に微小であり、そのユークリッド距離 $\sqrt{(p-p')^2 + (q-q')^2}$ はこれらの点の間の曲線に沿った距離について良好な近似を与える。一方、テーラー展開により、

$$q - q' \approx (dy/dx)|_{x=p} (p - p'), \quad (7)$$

なる近似式が得られる。いま、補間曲線上の点から曲線に沿って T_k だけ離れた点を求めるとすれば、式(7)を用いて次

式が得られる。

$$|p - p'| \sqrt{1 + \{(F_x / F_y) |_{x=p, y=q}\}^2} = T_k \quad (8)$$

ここで $F_x(p, q)$ および $F_y(p, q)$ は次式で定義される。

$$F_x(p, q) = 3A p^2 + 2C p q + D q^2 + 2E p + G q + H, \quad (9)$$

$$F_y(p, q) = 3B q^2 + C p^2 + 2D p q + 2F q + G p + I.$$

式(9)におけるAからIまでは式(2)に現われる係数であり、表1に各係数の対応関係および項数が示されている。これらの係数は (p, q) の存在するセグメントにおいて定義される。なお、数値計算を精度良く実行するため、 $|F_x(p, q)| > |F_y(p, q)|$ であれば、式(8)の代わりに次式を用いる。

$$|q - q'| \sqrt{1 + \{(F_y / F_x) |_{x=p, y=q}\}^2} = T_k \quad (10)$$

以下では、記述を簡単にするため式(8)を解くべき方程式であるとす。式(8)はパラメータ空間における隣接するサンプル点間距離 $|p - p'|$ と対応する補間曲線上(換言すれば表示空間上)での距離 T_k との関係をあてている。したがって、この方程式の未知数はパラメータ空間で定義される量である。いま、点 (p, q) が第 i セグメント上に存在するものとし、 p のパラメータ値を t_p とする。すなわち、 $p = a_3 t_p^3 + a_2 t_p^2 + a_1 t_p + a_0$ と仮定する。さらに p' のパラメータ値を $t_p + \Delta t$ とすれば、式(8)は Δt を未知数とする次の方程式となる。

$$a_3 \Delta t^3 + (3a_3 t_p + a_2) \Delta t^2 + (3a_3 t_p^2 + 2a_2 t_p + a_1) \Delta t - |M_i| = 0, \quad (11)$$

$$|M_i| = T_k \sqrt{1 + \{(F_x / F_y)_{x=p, y=q}\}^2}$$

この3次方程式を数値的に解くためには2分法が適している。この場合、初期値を T_k と仮定し、式(11)の左辺の

符号が互いに異なるような二つのパラメータ値を用いて計算を開始すれば速やかに収束解が得られる。 q' の値は Δt が求まれば式(1)より直ちに計算することができる。そこで、次のサンプル法を得る。

[サンプル法2]

3-1節で得られるサンプル点列の相対距離をもとに T_k ($k=1, 2, \dots, m$; m は生成するサンプル点数) を求める。この T_k を用いて方程式(11)を解くことによりサンプル点列をスプライン関数上に得る。

実際の数値計算アルゴリズムを図2に示す。まず、与えられたデータ点 P_1, P_2, \dots, P_n を3次スプライン補間し、各セグメントの係数を定める。次に、これらをもとにして曲線長および曲率を求め、サンプル点列 L_1, L_2, \dots, L_m を決定する。 L_1, L_2, \dots, L_m は、3-1節で述べたサンプル点列の相対距離を基にして、 L_1 と L_m との距離が補間曲線上において最初および最後のサンプル点間の距離に等しくなるように線形変換して得る。したがってこの点列は、実現すべきサンプル点列の曲線上での距離を与えている。ここまですべて3-1節の第1段階である。3-2節の段階は方程式(11)を解くことである。第1サンプル点を第1データ点(P_1)に一致させ、この x および y 座標をそれぞれ p および q として、この点から曲線上に沿って $T_1 = L_2 - L_1$ だけ離れた点のパラメータ値を計算する。得られる新サンプル点が第1サンプル点と同じセグメントにあればこれを第2サンプル点とする。新サンプル点と同じセグメント内になければ始点と反対側に隣接するデータ点、つまり P_2 を基準点として方程式(11)を解く。この手順を m 個のサンプル点が得られるまで繰り返す。

このアルゴリズムを用いて得られるサンプル点列は2種類の近似を導入して得られたものである。まず、前述のように方程式(11)は、式(7)に示されているように、パラメータ空間と表示空間との写像を線形近似して得られたものである。つぎに、アルゴリズムにおいては、曲線に沿う距離の算出はすべてユークリッド距離で近似している。これらの近似はいうまでもなくサンプル点数が増加し、サンプル点の間隔が小さくなるほど高精度となる。しかし、一般にはこの条件は満足されず、近似誤差の問題が生ずる。本手法を実用化するにあたり最大の問題は誤差累積による最終サンプル点のずれであると考えられる。すなわち、最終サンプル点を最終データ点に一致させようとしても、第1サンプル点より逐次的にサンプル点を計算してゆくと、最終サンプル点は一般

に最終データ点に一致しない。この問題を実用的見地から解決する最も簡便な方法としてつぎの方法を提案する。

まず、最終データ列の次に付加的なデータ点を1個導入する。この手法は、3次スプラインを計算する際、補間曲線の両端点において要求される付加的条件（本論文では自然スプラインを用いているのでこの条件は不要である）を得るためによく用いられる。具体的には、最終データ点 P_n に関して P_{n-1} に対称な位置にあるデータを生成すればよい。付加的なデータの導入により、 n 番目のセグメントが構成できる。こうすることにより、誤差累積により m 番目のサンプル点が P_n 以降に算出されても、式(1)において $t > t_{n-1}$ となるような不合理な演算を防ぐことができる。次に、最終サンプル点のパラメータ値を P_n のパラメータ値 (t_{n-1}) に一致させる。このためのパラメータ空間における線形補正を他のサンプル点についても施す。この方法は次節の適用例において用いられている。

4. 適用例

本論文における手法は(1)スプライン曲線上におけるサンプル点列の相対間隔を定める部分(3-1参照)、および得られたサンプル点列の相対位置を基にしてスプライン曲線上にサンプル点列を生成する部分(3-2参照)とに分割することができる。以下、両者とも用いた例、および、前者の処理を他の簡易法で置き換えた例を提示する。先ず、図3に適用例の基礎となるスプライン補間曲線を示す。始点は「8字型」曲線の交点上とする。図に示すように19個のデータ(X印)を3次スプライン曲線(自然スプライン)で補間した曲線について考察する。ここで、始点、第10データ、および第19データは同じ位置にある。また、前節で述べた、累加計算誤差による最終サンプル点が不合理な位置に計算される問題を解決するため、第20データを付加してあり、このデータは第2データの位置に一致させてある。補間曲線を形成している18個のセグメント(付加的なセグメントは除く)上に提案手法によりサンプル点列を生成する。

図4に41個から成る等間隔サンプル点列(+印)の例を示す。等間隔サンプル点列は3-1の処理を施すことなく、任意の、相対距離一定の41個からなるサンプル点データで計算できる。図から分かるように、曲率の大きな部分のサンプル点間隔がやや広がっているが、全体的にはほぼ曲線に沿って等間隔なサンプル点列が生成されている。なお、累積計算誤差のため、以下の例でもそうであるが、最初および最

後のサンプル点の中心に来るべきサンプル点(この例では第21データ点)の位置が曲線の交点からややずれている。

図5は3-1および3-2の処理を適用して得られた41個のサンプル点列を示している。図3のスプライン関数に沿う曲率Kの変化は図1に表示してある。第一ステップとしてKのサンプル間における累加積分値が一定になるようにして方程式(11)を解き、個々のサンプル点の相互距離を得た。21番目のサンプル点において累加計算誤差による位置ずれが生じているが、直線部分は高速で、曲率の大きな部分は低速で移動する物体の動きがシミュレートされている。しかし、最終サンプル点は前節で述べた処理により最終データに一致させてある。

図6は図5と異なる物体の動きのシミュレーション例として3-1の処理を用いずに、サンプル点の相対位置を関数 $\sum_{i=1}^m |\cos \pi i / 21|$, ($m=1, 2, \dots, 42$) によって定めたものである。これにより得られる42個のサンプル点の相対位置と第1サンプル点を加えた合計43個のサンプル点を図に示すように計算した。この例では「8字型」形状の交点から最遠点付近で、物体の動きが非常に小さくなっている。

5. 結言

与えられた3次スプライン関数上に、任意の生成規則に基づいて得られる相対距離を有する点列を、サンプル点列とし

表1 方程式(2)の各係数

記号	変数	係数の項数
A	x^3	1
B	y^3	1
C	x^2y	1
D	xy^2	1
E	x^2	7
F	y^2	7
G	xy	8
H	x	21
I	y	21
J	定数項	34

て実現する方法について述べた。得られたアルゴリズムには二種類の近似を用いているため、累加計算誤差という問題がある。しかし、最終サンプル点を補間曲線上での最終点（一般には最終データ点）に一致させるという簡便な方法を導入することにより、コンピュータアニメーションなどの実用上は提案手法を用いても問題ないと考えられる。

本論文で得られたサンプル点列配置法は、直接的にはコンピュータアニメーションにおける動作シミュレーションに有効であるが、提案手法はパラメトリック表現された関数を利用して現象を記述しなければならないような問題一般に適用することが可能である。

参考文献

[1] Rogers, D F and Adams, J A , "Mathematical elements for computer graphics," McGraw-Hill (1976)
 [2] de Boor, C, A practical guide to splines, Springer Verlag (1978)
 [3] Kochanek, D H U and Bartels, R H "Interpolating

splines with local tension, continuity, and bias control," Comput. Gr. (July 1984) Vol.18, No.3 pp.33-41

[4] 例えば Haigh, M J, An Introduction to Computer-Aided Design and Manufacture, Blackwell Sci. Pub. (1985)

[5] Brodlie, K W "A review of methods for curve and function drawing," (in Mathematical Methods in Computer Graphics and Design, K.W.Brodlie ed.) Academic Press (1980)

[6] Steketee, S N and Badler, N I "Parametric Keyframe Interpolation Incorporating Kinetic Adjustment and Phrasing Control' Comput. Gr. (1985) Vol.19, No.3 pp. 255-262

[7] Harada, K and Nakamae, E, "Cubic Point Extrapolation," Trans. IECE Japan (1983) Vol. E66, No.1, pp. 28-33

[8] Sederberg, T W and Anderson, D.C., "Implicit representation of parametric curves and surfaces," Comput. Vis., Graph. and Image Proc. (1984) Vol.28, pp.72-84

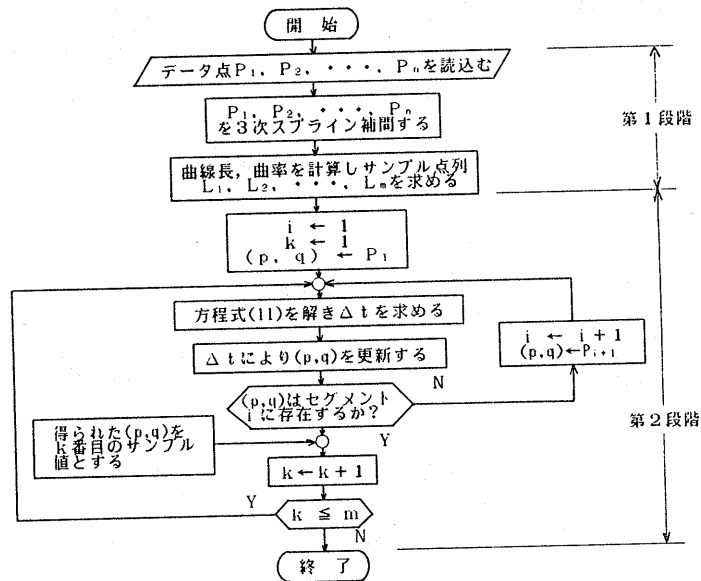


図2 提案アルゴリズム

図3 スプライン補間曲線

19個のデータに付加的なデータ1個を付け加えたものを自然スプラインで補間している。

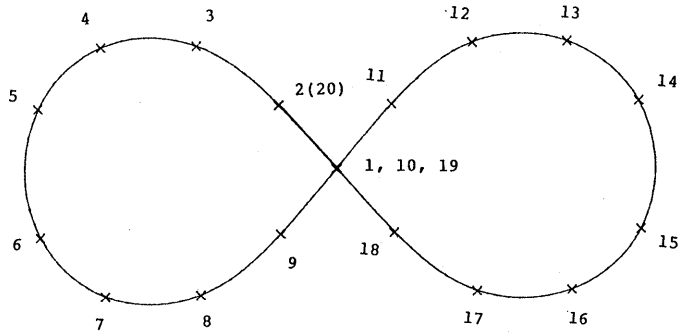


図4 適用例1

等速度運動のシミュレーション

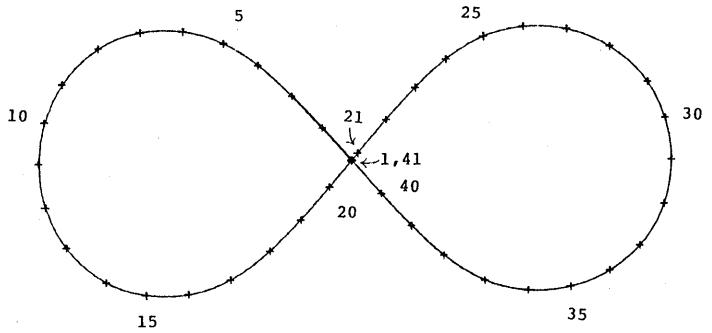


図5 適用例2

図1の曲率変化を基にして計算した運動のシミュレーション

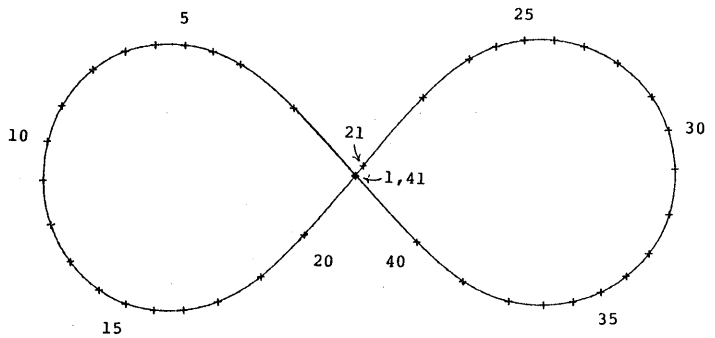


図6 適用例3

関数 $\sum_{i=1}^m |\cos \pi i/21|$, ($m=1, 2, \dots, 42$) によって生成された動きのシミュレーション

