

ガウス球上の細分割曲線・曲面と そのストリームラインモデリングへの応用

三浦 憲二郎†

曲率や曲率の変化率は曲面の滑らかさや美しさ (fairness) を決定する本質的な要因である。しかしながら、これまでの曲面表現式のほとんどはこれらの特性を直接的に操作する方法を設計者やデザイナーに提供していない。最近提案された曲面デザインの 1 つの方法論であるストリームラインモデリング (streamline modeling) は、ユーザが接線ベクトルを制御する (したがって、より直接的に曲率や曲率の変化率を制御する) ことを可能にしこの欠点を克服することを意図している。曲面の等パラメータ曲線をこれまでのように位置の混ぜ合わせにより定義するのではなく、接線の混ぜ合わせとしてそれら (ストリームライン) を定義し、ストリームラインの集合として曲面を定義する。この新しい曲面デザイン方法論は高品質な曲面を生成できるが、接線の混ぜ合わせに多大な処理時間を必要とする。この論文では接線ベクトルの混ぜ合わせ手法のための細分割 (subdivision) を提案する。この手法によりガウス球上の曲線や曲面を生成するために Chaikin や B-spline, Kobbelt 型の細分割を行うことができる。さらに、この新しい接線ベクトルの混ぜ合わせ手法に基づいた新しいストリームラインモデリングの技法を提案する。この技法はストリームラインに必要とされる数値積分のための処理時間を削減する。新しい技法に基づくプロトタイプを開発し、ストリームラインモデリングを用いた曲面設計が実時間で行えることを確認した。

Subdivision Curves and Surfaces on the Gaussian Sphere and Their Applications to the Streamline Modeling

KENJIRO T. MIURA†

Curvature and variation of curvature are the essential factors in determining the fairness of a surface. Unfortunately, most of the traditional surface representation schemes do not provide users with direct manipulation techniques of these quantities. *Streamline modeling*, a recently proposed free-form surface design methodology, is aimed at overcoming this shortcoming by allowing a user to control tangent vectors (and, consequently, curvature and variation of curvature) of the surface to be designed directly. A free-form surface is regarded as a set of streamlines: iso-parametric lines defined by blending directions of tangent vectors instead of blending positions of control points. This new surface design methodology can generate high quality smooth surfaces but requires much processing power for tangent vector blending. In this paper, we present subdivision based blending techniques of tangent vectors. These techniques can be used to develop subdivision techniques for curves and surfaces on the Gaussian sphere, such as Chaikin, B-spline, and Kobbelt subdivisions. We also present new streamline modeling techniques based on the new tangent vector blending techniques. The new techniques reduce the processing time for the integration process required in streamline modeling. A prototype system based on the new techniques shows that free-form surface design using the streamline modeling methodology can achieve real-time performance.

1. はじめに

ストリームラインモデリング (streamline modeling) は自由曲面のモデリング手法の 1 つであり、曲面の流れ (streamline) を強く意識したデザイン手法である。三浦は制御接線の混ぜ合わせ (blending) によ

り単位 4 元数積分 (quaternion integral: QI) 曲線・曲面^{1),3),5)} を提案するとともに、QI 曲面を用いたストリームラインモデリング²⁾ を提案した。ストリームラインモデリングでは、高品質な曲面生成をするために位置の指定とそこでの法線の指定を分離して別々に行うのではなく (たとえば、文献 6) 参照)、それらを接線方向の指定を主体として同時に行う。たとえば、位置の指定そのものに無理がある場合 (極端な例としては、ランダムに与えた点列を内挿する場合) に美し

† 静岡大学工学部機械工学科
Department of Mechanical Engineering, Shizuoka University

い曲面を生成することは困難であり、位置とそこでの法線（実際には接線方向）を同時に対話的に変化させながら曲面をデザインする点にストリームラインモデリングの方法論としての新しさ、利点がある。

QI 曲面を用いたストリームラインモデリングでは、各ストリームラインの接線方向は制御接線の混ぜ合わせにより指定され、接線方向を積分することにより曲面の形状が一意に得られる。QI 曲面は曲率や曲率変化率が穏やかに変化する滑らかな曲面を生成することに適している⁴⁾が、形状の計算に数値積分を必要とし複雑な曲面を会話的にデザインするには生成処理の高速化が望まれる。

そこで、本研究では細分割曲面 (subdivision surface) の考え方をガウス球 (単位球) 上に定義される曲面に応用し、接線方向の混ぜ合わせ計算の高速化と曲面の表示処理の高速化を行った。さらに、これらの手法に基づく曲面モデリングシステムを開発した。このシステムでは曲面を任意の粗さ (細分割曲面の分割深さ) で表示できるので会話を損なうことなく自由曲面をデザインできる。

以下の説明では、ガウス球上の曲線・曲面と接線方向を積分して生成される曲線・曲面とを明確に区別するために、後者の曲線・曲面をターゲット曲線・曲面と呼ぶ。

2. ストリームラインモデリングの基本概念

ストリームラインモデリングの基本的な考え方は、図 1 に示したようにストリームライン (streamline) と呼ぶ等パラメータ曲線の集合としてパラメトリック曲面を作成することである。各々のストリームラインは一定のパラメータ値 t_0 を持ち、 $S(s_0, t_0)$ として与えられた点からパラメータ s の正、あるいは負の方向に伸びていく。図 1 (a) に示したように、 s_0 は s の最小値 s_{\min} や最大値 s_{\max} に一致させることも、図 1 (b) のようにそれらの中間値とすることもできる。

曲面 $S(s, t)$ に対して、弧長パラメータ s に関する 1 次微分 $\partial S(s, t)/\partial s = s(s, t)$ が与えられていれば、曲面は次式により定義される：

$$S(s, t) = S(s_0, t) + \int_{s_0}^s s(s, t) ds. \quad (1)$$

ここで、 s_0 は s パラメータ空間の固定された 1 点であり、曲線 $S(s_0, t)$ を初期曲線と呼ぶ。式 (1) はパラメータ s が $s = s_0$ を固定して得られるパラメータ t 方向の初期曲線と s に関する 1 次微分ベクトル $s(s, t)$ がパラメータ空間内のすべての領域で与えられていれば、式 (1) が示すように 1 次微分ベクトル

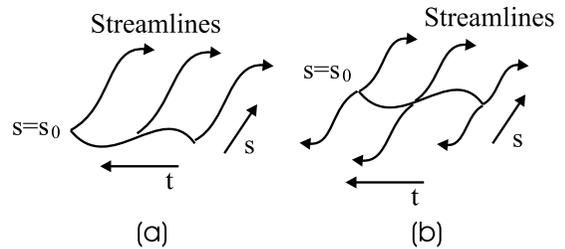


図 1 等パラメータ曲線としてのストリームライン
Fig. 1 Streamlines as iso-parametric curves.

を積分することによって曲面が構築できる。 $s(s, t)$ の大きさ (ノルム) をつねに 1 となるように制限すれば ($|s(s, t)| = 1$)、パラメータ s は弧長となり、各々のストリームラインの長さは s の最大値にのみ依存する。 s の最大値は t の関数とすることもできる⁴⁾。

式 (1) は t に関する 1 次微分 $\partial S(s, t)/\partial t = t(s, t)$ が次式で与えられることを示している：

$$t(s, t) = \frac{\partial S(s, t)}{\partial t} = t_0 + \int_{s_0}^s \frac{\partial s(s, t)}{\partial t} ds. \quad (2)$$

ここで $t_0 = \partial S(s_0, t)/\partial t$ である。 $t(s, t)$ のノルムはつねに 1 であるとは限らないので、 s は弧長であるが t は一般的に弧長ではない。

さらに、 $s(s, t)$ のノルムをつねに 1 となるように制限すれば式 (1) における $s(s, t)$ はガウス球 (3 次元空間内の単位球) 上の曲面であることに注意する。その曲面上の点はターゲット曲面のパラメータ s 方向の接線ベクトルに対応している。これまで述べたように、ストリームラインモデリングにおける自由曲面の設計法では $s(s, t)$ をいかに指定するかが重要であり、その指定法としての接線の混ぜ合わせの概念が重要となる。

3. ガウス球上の曲線

ストリームラインモデリングにおいては、接線ベクトルの内挿法がターゲット曲面の良否を左右するだけでなくその表示や変形のための処理時間の長短を決める鍵となる。したがって、ストリームラインモデリングにおける基本問題の 1 つは、ガウス球上に滑らかな曲線を生成するために、いかに与えられた接線ベクトルを近似、あるいは内挿するかである。この章では、接線ベクトルの内挿法の問題点を明らかにするとともに

Bézier や B-spline 曲線・曲面の指定法として制御点の混ぜ合わせの概念が合理的であることと同様に、接線の混ぜ合わせが合理的であり重要である。

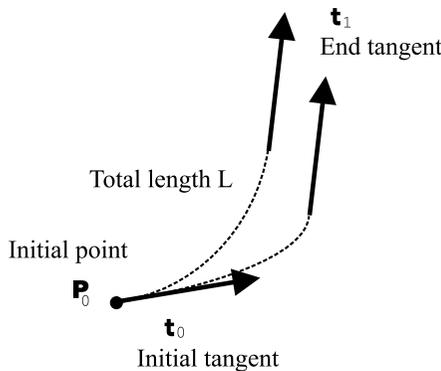


図 2 内挿問題: 始点と、始点と終点での接線ベクトルの方向、全長が満足すべき条件として与えられているとき、それを満たす 3 次元空間曲線を見つけよ。

Fig.2 Interpolation problem: finding a smooth curve whose initial point, tangent vectors at its initial and end points and total length are given.

に、その問題点を再分割の概念を使って解決する手法を述べる。

3.1 接線ベクトルの内挿

接線ベクトルの一般的な混ぜ合わせ法(任意個の接線ベクトルを混ぜ合わせる方法)を考える前に、接線ベクトルの混ぜ合わせにもなう問題点を明らかにするために以下のような幾分技巧的ではあるが簡単な内挿問題を考える(図 2 参照)。

始点と、始点と終点での接線ベクトルの方向、全長が満足すべき条件として与えられているとき、それを満たす 3 次元空間曲線を見つけよ。

この問題の最も自然な望ましい曲線は円弧である。というのは、接線ベクトルの長さを一定に保ち、始点での接線ベクトルを終点の接線ベクトルへ、始点と終点での接線ベクトルに直交するベクトル周りに、弧長に比例した角度回転させれば結果として円弧が得られる。円弧の曲率はつねに一定であり、接線ベクトルの変化率はつねに同一点(円弧の中心)を向いている。

上の議論から接線ベクトルの混ぜ合わせに回転を考慮することが重要であることが分かる。以後、表記上の簡便さから任意軸周りの回転を単位 4 元数 (unit quaternion) を使って表す。与えられた単位 4 元数 q に対して、3 次元空間内の回転 R_q は次式により定義される:

$$R_q(v) = qvq^{-1}, \quad \text{for } v \in R^3.$$

ここで $v = (x, y, z)$ は 4 元数 $(0, x, y, z)$ と解釈し、積は 4 元数の乗算を用いる。

与えられた 2 つの単位接線ベクトル $t_0 = q_0 \hat{v}_0 q_0^{-1}$

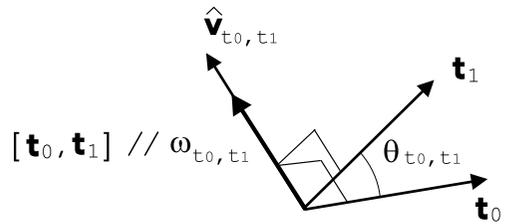


図 3 回転軸
Fig.3 The axis of rotation.

と $t_1 = q_1 \hat{v}_1 q_1^{-1}$ (q_0 と q_1 は単位 4 元数) に対して、2 つの単位ベクトル t_0 と t_1 の終点を結ぶガウス球上の測地線 $\gamma_{t_0, t_1}(u)$ は、

$$\gamma_{t_0, t_1}(u) = q_{t_0, t_1}(u) t_0 q_{t_0, t_1}^{-1}(u)$$

と与えられる。ここで、

$$q_{t_0, t_1}(u) = \exp(\omega_{t_0, t_1} u)$$

である。 ω_{t_0, t_1} はパラメータ u を時間を見なせば回転の角速度に一致する量であり、図 3 に示したようにその方向は t_0 と t_1 の外積に一致する。 ω_{t_0, t_1} は、

$$\begin{aligned} \omega_{t_0, t_1} &= \frac{1}{2} \sin^{-1}(|t_0 \times t_1|) \frac{t_0 \times t_1}{|t_0 \times t_1|} \\ &= \frac{1}{2} \theta_{t_0, t_1} \hat{v}_{t_0, t_1} \end{aligned} \quad (3)$$

と与えられる。ただし、 $\sin^{-1}(|t_0 \times t_1|)$ の値は t_0 と t_1 の角度 θ_{t_0, t_1} となるようその値域を定め、 $|t_0 \times t_1|$ は $t_0 \times t_1$ の大きさとする。 \hat{v}_{t_0, t_1} はその方向が $t_0 \times t_1$ に一致する単位ベクトルである。

式 (3) の ω_{t_0, t_1} は $\gamma_{t_0, t_1}(u)$ が $\gamma_{t_0, t_1}(0) = t_0$ と $\gamma_{t_0, t_1}(1) = t_1$ を満足させる。接線ベクトルが $q_{t_0, t_1}(u) t_0 q_{t_0, t_1}^{-1}(u)$ で与えられる曲線は円弧である。さらに、 q_0 と q_1 が t_0 と t_1 に対して一意に定まらないのに対して、 ω_{t_0, t_1} は t_0 と t_1 により一意に定まる。もし、slerp (spherical interpolation) ⁷⁾ を計算するように $\omega_{t_0, t_1} = \log(q_0^{-1} q_1)$ を用いると得られる曲線は円弧になるとは限らない。

以上の議論により、ストリームラインモデリングのための接線ベクトルの軌跡となるガウス球上の曲線を生成するのに接線ベクトルの回転が有効であり、接線の単純なベクトル和とその単位ベクトル化が不適當であることが分かる。Kim ⁷⁾ らの提案する単位 4 元数曲線を基礎として、三浦 ⁵⁾ はストリームラインモデリングのために次に示す単位 4 元数曲線を提案した:

与えられた接線ベクトル t に対して、 $t = q \hat{v}_0 q^{-1}$ を満たす q は一意には決まらず、無数の回転が \hat{v}_0 を t に移動することに注意する。 $\hat{v}_0 = (1, 0, 0)$ とすると、たとえば $q = \cos \beta + \hat{v} \sin \beta = (\cos \beta, 0, \cos \alpha \sin \beta, \sin \alpha \sin \beta)$ として与えられる。

$$q(u) = \left(\prod_{i=n}^1 \exp(\omega_i \tilde{B}_{i,k}(u)) \right) q_0. \quad (4)$$

ここで、 $\tilde{B}_{i,k}(u)$ は B-spline 基底関数 $B_{i,k}$ の累積形式であり、

$$\begin{aligned} \tilde{B}_{i,k}(u) &= \sum_{j=1}^n B_{j,k}(u) \\ &= \begin{cases} \sum_{j=i}^{i+k-1} B_{j,k}(u) & \text{if } u_i < u < u_{i+k-1} \\ 1 & \text{if } u \geq u_{i+k-1} \\ 0 & \text{if } u \leq u_i. \end{cases} \end{aligned} \quad (5)$$

である。式 (4) の ω_i は、

$$\omega_i = \frac{1}{2} \sin^{-1}(|t_{i-1} \times t_i|) \frac{t_{i-1} \times t_i}{|t_{i-1} \times t_i|}$$

で与えられる。 ω_i はグローバル座標系で定義される制御接線ベクトルから算出されるので、グローバル座標系で意味を持ち、その点でも Kim ら⁷⁾ の定義法と異なる。接線が $q(u)v_0q(u)^{-1}$ で与えられる曲線が単位 4 元数積分 (QI) 曲線である⁵⁾。

3.2 接線ベクトルの線形結合

前節で述べた接線ベクトルの混ぜ合わせ法によりガウス球上で滑らかな曲線が生成されるが、以下の 2 つの重大な問題が未解決である。

1 つは制御接線ベクトルの指定順序を入れ換えるに対応するパラメータ値に対する接線方向が異なってしまうことである。すなわち、単位 4 元数曲線 $q_0(u)$ が接線ベクトル t_0, t_1, \dots, t_n により定義されていることを以下のように表すと、

$$\begin{aligned} q_0(u) &= q_0[t_0, t_1, \dots, t_n](u) \\ &= \left(\prod_{i=n}^1 \exp(\omega_i \tilde{B}_{i,k}(u)) \right) q_0. \end{aligned} \quad (7)$$

接線ベクトルの指定順序を反転した q_r は、

$$\begin{aligned} q_r(u) &= q_r[t_n, t_{n-1}, \dots, t_0](u) \\ &= \left(\prod_{i=1}^n \exp(-\omega_i \tilde{B}_{i,k}(u)) \right) \left(\prod_{i=n}^1 \exp(\omega_i) \right) q_0 \end{aligned} \quad (8)$$

と表せる。このとき一般的に、

$$q_r(u) \neq q_0(1-u) \quad (9)$$

となる。例として、図 4 に B-spline 型単位 4 元数曲線 (階数 3) の制御接線の指定順序を反転した場合の形状の違いを示す。

もう 1 つの問題は単位 4 元数曲線を基礎とするストリームラインの計算は強力な計算機パワーを必要とすることである。単位 4 元数曲線の積分を解析的に行う

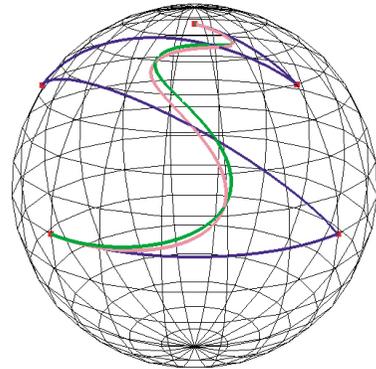
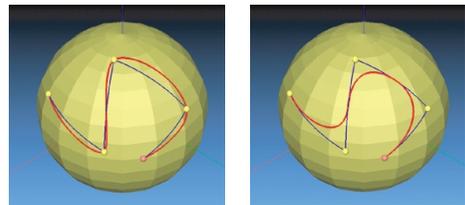


図 4 制御接線の指定順序反転による形状の変化：順方向 (赤線)，逆方向 (緑線)

Fig. 4 Shape difference by reversing tangent order: original (red), reversed (green).



(a) 4 点内挿細分割曲線 (赤線, $\omega = 1$, 分割深さ = 3) (b) B-spline 単位 4 元数曲線⁵⁾ (赤線)

図 5 ガウス球面上の曲線

Fig. 5 Curve on the Gaussian sphere.

ことはできず数値積分が不可避であり、指数関数を含む数値積分に多大な処理時間が必要である。

これらの問題を解決するために、ガウス球上で細分割曲線・曲面を定義し、それらのガウス球上の曲線・曲面上の 1 点を 1 つの接線方向に対応させその方向をその点のパラメータ値でのターゲット曲線・曲面の接線方向とする。ガウス球上の点列や点群に対して細分割を繰り返すことによりガウス球面上に C^0 や C^1 連続性を満たす滑らかな曲線・曲面を生成する。細分割の概念を S^2 空間 (ガウス球上の空間) に適応させるために、接線ベクトルの線形結合を定義する必要がある。ここで述べる「接線ベクトルの線形結合」は S^2 空間を線形空間とする演算を定義することではなく、順序付けされた複数の単位接線ベクトルとその重み定数から 1 つの単位ベクトルを算出する「結合」操作を定義することを意味する。

R^3 空間においては 2 点 p_0 と p_1 の中点 p_m は、

$$p_m = \frac{1}{2}(p_0 + p_1) \quad (10)$$

で計算される。 S^2 空間では、2 つの接線 t_0 と t_1 の

中間接線 t_m を次式で定義することはできない:

$$t_m = \frac{1}{2}(t_0 + t_1). \quad (11)$$

なぜならば, t_0 と t_1 の大きさが 1 であっても t_m の大きさは必ずしも 1 にならないからである. 前節の議論をふまえ,

$$t_m = q\left(\frac{1}{2}\right) t_0 q^{-1}\left(\frac{1}{2}\right) \quad (12)$$

と定義すべきである. ここで,

$$q(u) = \exp(\omega_{t_0, t_1} u) \quad (13)$$

であり, ω_{t_0, t_1} は式 (3) により与えられる. 定義は式 (12) であるが, 実際の中間接線の計算に 4 元数を用いる必要はなく, 2 つの接線のベクトル和を計算しそれを単位化すればよいので高速に実行することができる.

中間接線の算出法を一般化し, 2 つのベクトル t_0 と t_1 の線形結合を以下のように定義する:

$$\begin{aligned} t(u) &= (1-u)t_0 \oplus ut_1 \\ &= q(u)t_0 q^{-1}(u). \end{aligned} \quad (14)$$

上式は交換則が成り立つ (commutative), すなわち

$$(1-u)t_0 \oplus ut_1 = ut_1 \oplus (1-u)t_0. \quad (15)$$

であること, しかしながら結合則は成り立たないことに注意する.

3.3 接線ベクトルの線形結合の一般形

2 つの接線ベクトルの混ぜ合わせを表す単位 4 元数曲線 $q_o(u)$ はそれらのベクトルの指定順序を反転した単位 4 元数曲線 $q_r(u)$ と $q_r(u) = q_o(1-u)$ の意味で一致する. 同様に, 固定された軸回りの回転は, 複数の回転であってもそれらの順序に結果は依存せず, 元の単位 4 元数曲線とその反転が一致する. しかしながら, 回転軸が固定されない一般的な場合は一致しない. この節では, この一般的な場合に反転を一致させる方法を提案する.

まず, 3 次元空間内の点の線形結合について考える. 3 次元空間の $n+1$ 点, $p_i, i=0, \dots, n \in R^3$ の一般的な線形結合 p は,

$$p = c_0 p_0 + c_1 p_1 + \dots + c_n p_n \quad (16)$$

で定義される. ここで, $c_i, i=0, \dots, n$, は係数である. 提案する方法は n が偶数であるか, 奇数であるかに依存するので, $n=2, 3$ の場合について論じ, その後で一般の場合を述べる.

$n=2$ であれば, 式 (16) は次式となる:

$$\begin{aligned} p &= p[p_0, p_1, p_2] \\ &= c_0 p_0 + c_1 p_1 + c_2 p_2. \end{aligned} \quad (17)$$

アフィン変換に対する不変性の要求のため, 細分割において係数 c_i の総和 $\sum_0^n c_i$ は 1 となる. したがって, $c_0 + c_1 + c_2 = 1$ が成り立つ. 総和が 1 であるこ

と指定順序の反転についての要求を考慮し, 式 (17) を以下のように変形する:

$$\begin{aligned} p &= \left(c_0 p_0 + \frac{1}{2} c_1 p_1\right) + \left(\frac{1}{2} c_1 p_1 + c_2 p_2\right) \\ &= \left(c_0 + \frac{1}{2} c_1\right) \left(\frac{c_0}{c_0 + \frac{1}{2} c_1} p_0 + \frac{\frac{1}{2} c_1}{c_0 + \frac{1}{2} c_1} p_1\right) \\ &\quad + \left(\frac{1}{2} c_1 + c_2\right) \left(\frac{\frac{1}{2} c_1}{\frac{1}{2} c_1 + c_2} p_1 + \frac{c_2}{\frac{1}{2} c_1 + c_2} p_2\right). \end{aligned} \quad (18)$$

たとえば, $c_0 = 1/7, c_1 = 2/7, c_2 = 4/7$ であれば,

$$p = \frac{2}{7} \left(\frac{1}{2} p_0 + \frac{1}{2} p_1\right) + \frac{5}{7} \left(\frac{1}{5} p_1 + \frac{4}{5} p_2\right) \quad (19)$$

である. これを接線ベクトル $t_i \in S^2, i=0, 1, 2$, に置き換えるとそれらの線形結合 t は二項演算子 \oplus を繰り返し使い,

$$\begin{aligned} t &= t[t_0, t_1, t_2] \\ &= c_0 t_0 \oplus c_1 t_1 \oplus c_2 t_2 \\ &\equiv \left(c_0 + \frac{1}{2} c_1\right) \left(\frac{c_0}{c_0 + \frac{1}{2} c_1} t_0 \oplus \frac{\frac{1}{2} c_1}{c_0 + \frac{1}{2} c_1} t_1\right) \\ &\quad \oplus \left(\frac{1}{2} c_1 + c_2\right) \left(\frac{\frac{1}{2} c_1}{\frac{1}{2} c_1 + c_2} t_1 \oplus \frac{c_2}{\frac{1}{2} c_1 + c_2} t_2\right). \end{aligned} \quad (20)$$

この形式であれば, 係数 c_i が u の関数であり, すなわち, $c_0 = c_0(u), c_1 = c_1(u), c_2 = c_2(u)$, さらに $c_2(u) = c_0(1-u), c_1(u) = c_1(1-u)$ であれば (これらの条件は, たとえば, 係数が Bernstein 基底関数であれば満たされる),

$$t_o = t_o[t_0, t_1, t_2](u) \quad (21)$$

$$t_r = t_r[t_2, t_1, t_0](u) \quad (22)$$

に対して,

$$t_r(u) = t_o(1-u) \quad (23)$$

が成り立つ.

式 (16) において n が 3 の場合は, $n=2$ の場合よりも簡潔に変形され,

$$\begin{aligned} p &= c_0 p_0 + c_1 p_1 + c_2 p_2 + c_3 p_3 \\ &= (c_0 + c_1) \left(\frac{c_0}{c_0 + c_1} p_0 + \frac{c_1}{c_0 + c_1} p_1\right) \\ &\quad + (c_2 + c_3) \left(\frac{c_2}{c_2 + c_3} p_2 + \frac{c_3}{c_2 + c_3} p_3\right) \end{aligned} \quad (24)$$

となる. $n=2$ の場合と同様に, 接線ベクトル $t_i \in S^2, i=0, \dots, 3$, に置き換えそれらの線形結合 t を二項演算子 \oplus を繰り返し使い,

$$\begin{aligned} t &\equiv (c_0 + c_1) \left(\frac{c_0}{c_0 + c_1} t_0 \oplus \frac{c_1}{c_0 + c_1} t_1\right) \\ &\quad \oplus (c_2 + c_3) \left(\frac{c_2}{c_2 + c_3} t_2 \oplus \frac{c_3}{c_2 + c_3} t_3\right) \end{aligned} \quad (25)$$

により定義する. この場合も接線ベクトルの順序の反

転に関して $n = 2$ と同様な議論が成り立つ．

n が一般の場合は，まず n が偶数であれば $n = 2$ と同様な方法で以下のように全体を 2 つのグループに分け，

$$\begin{aligned} t &= c_0 t_0 \oplus c_1 t_1 \oplus \cdots \oplus c_n t_n \\ &= \left(c_0 t_0 + \cdots + \frac{1}{2} c_{\frac{n}{2}} t_{\frac{n}{2}} \right) \\ &\quad \oplus \left(\frac{1}{2} c_{\frac{n}{2}} t_0 + \cdots + \frac{1}{2} c_n t_n \right) \end{aligned} \quad (26)$$

n が奇数であれば $n = 3$ と同様に，

$$\begin{aligned} t &= c_0 t_0 \oplus c_1 t_1 \oplus \cdots \oplus c_n t_n \\ &= \left(c_0 t_0 \oplus \cdots \oplus c_{\frac{n-1}{2}} t_{\frac{n-1}{2}} \right) \\ &\quad \oplus \left(c_{\frac{n+1}{2}} t_{\frac{n+1}{2}} \oplus \cdots \oplus \frac{1}{2} c_{\frac{n}{2}} t_n \right) \end{aligned} \quad (27)$$

と変形する．この変形を左右のグループに対してその項数の偶数，奇数に応じて上記のグループ化を 2 項の演算になるまで再帰的に適用することによって接線ベクトルの一般的な線形結合が定義できる． \oplus 演算子を挟む 2 つの係数の和がつねに 1 であることに注意する．たとえば，式 (20) において最初の \oplus を挟む和が 1 になることは自明であり，第 2 番目の \oplus の左右の 2 項の和は，

$$\left(c_0 + \frac{1}{2} c_1 \right) + \left(\frac{1}{2} c_1 + c_2 \right) = c_0 + c_1 + c_2 \quad (28)$$

でありアフィン変換に対する不変性の要求のため 1 となる．このことにより \oplus を適切に施すことが保証される．

この節で述べた定義を用いれば，一般的な接線ベクトルの混ぜ合わせであっても 4 元数の演算をしないで回転計算として計算可能なので，その回転計算の繰返しにより高速に算出することができる．

3.4 細分割曲線への応用

この節では，これまでに述べた接線ベクトルの内挿法を典型的な再分割曲線の定義法に対応させ，それらの処理速度を比較する．

3.4.1 Chaikin のアルゴリズム

Chaikin のアルゴリズム⁸⁾ は繰返しコーナカッティングをポリゴンに施すことにより 2 次 B-spline 曲線を生成する．1 回の細分割はポリゴンの辺の $(1/4, 3/4)$ の位置に新たな点を生成する．細分割の深さ j における $n+1$ 個の頂点 p_i^j , $i = 0, \dots, n$ を持つポリゴンに対して，ポリゴンの辺 $p_i^j p_{i+1}^j$ に以下の 2 点を新たに生成する：

$$p_i^{j+1} = \frac{3}{4} p_i^j + \frac{1}{4} p_{i+1}^j, \quad (29)$$

$$p_{i+1}^{j+1} = \frac{1}{4} p_i^j + \frac{3}{4} p_{i+1}^j. \quad (30)$$

Chaikin のアルゴリズムは新しい点の生成に 2 点しか用いず，これをガウス球上の曲線に拡張するのは容易であり， \oplus を用いて，

$$t_i^{j+1} = \frac{3}{4} t_i^j \oplus \frac{1}{4} t_{i+1}^j, \quad (31)$$

$$t_{i+1}^{j+1} = \frac{1}{4} t_i^j \oplus \frac{3}{4} t_{i+1}^j. \quad (32)$$

により定義できる．

式 (31) は以下のように，

$$t_{2i}^{j+1} = \frac{1}{2} t_i^j \oplus \frac{1}{2} \left(\frac{1}{2} t_i^j \oplus \frac{1}{2} t_{i+1}^j \right) \quad (33)$$

に書き換えることができ，2 つのベクトルの和とその単位ベクトル化を 2 回繰り返すことにより計算できる．式 (32) も同様な変形ができ，高速に接線ベクトルすることができる．

3.4.2 B-spline 細分割

B-spline 細分割は Chaikin のアルゴリズムの一般化であり，精緻化方程式 (refinement equation)⁹⁾ に従う．次数 l の B-spline の精緻化方程式は，

$$B_l(t) = \frac{1}{2^l} \sum_{k=0}^{l+1} \binom{l+1}{k} B_l(2t-k). \quad (34)$$

2 次 B-spline 細分割は Chaikin のアルゴリズムと一致し，3 次 B-spline 細分割は，

$$p_{2i}^{j+1} = \frac{1}{8} (p_i^j + 6p_{i+1}^j + p_{i+2}^j), \quad (35)$$

$$p_{2i+1}^{j+1} = \frac{1}{2} (p_{i+1}^j + p_{i+2}^j) \quad (36)$$

で与えられる．これをガウス球上の曲線に拡張すれば，

$$\begin{aligned} t_{2i}^{j+1} &= \frac{1}{2} \left(\frac{1}{4} t_i^j \oplus \frac{3}{4} t_{i+1}^j \right) \\ &\quad \oplus \frac{1}{2} \left(\frac{3}{4} t_{i+1}^j \oplus \frac{1}{4} t_{i+2}^j \right), \end{aligned} \quad (37)$$

$$t_{2i+1}^{j+1} = \frac{1}{2} t_{i+1}^j \oplus \frac{1}{2} t_{i+2}^j \quad (38)$$

と定義される．Chaikin のアルゴリズムと同様，式 (37) を計算するには，2 つのベクトルの和とその単位ベクトル化を 5 回繰り返すことによって，また式 (38) は 1 回施すことによって高速に算出することができる．

3.4.3 4 点内挿細分割

4 点内挿細分割は，以下のようにポリゴンを分割し，

$$p_{2i}^{j+1} = p_i^j \quad (39)$$

$$p_{2i+1}^{j+1} = \frac{8+\omega}{16} (p_i^j + p_{i+1}^j) - \frac{\omega}{16} (p_{i-1}^j + p_{i+2}^j) \quad (40)$$

$0 < \omega < 2(\sqrt{5}-1)$ であることが滑らかな曲線へ収束するための十分条件である¹⁰⁾．標準値は $\omega = 1$ で

表 1 計算時間の比較 (Pentium III 600 MHz)

Table 1 Comparison of processing time (Pentium III 600 MHz).

Curve	order	# c-tan.	# tan.	time (ms)
Sub. (Chaikin)	5	386 (depth=7)		0.31
Sub. (Cubic B-s.)	5	259 (depth=7)		0.63
Sub. (4 int.)	5	257 (depth=6)		1.09
B-spline	3	5	257	4.69
B-spline	4	5	257	4.84
B-spline	5	5	257	5.16

ある。

これは次式により拡張できる： S^2 空間に

$$t_{2i+1}^{j+1} = \frac{1}{2} \left(-\frac{\omega}{8} t_{i-1}^j \oplus \frac{8+\omega}{8} t_i^j \right) \oplus \frac{1}{2} \left(\frac{8+\omega}{8} t_{i+1}^j \oplus -\frac{\omega}{8} t_{i+2}^j \right). \quad (41)$$

1 を超える係数と負の値をとる係数が存在するがそれらの和は 1 であり、回転角度を 2 つの接線ベクトル間に限定しなければ計算可能である。図 5 (p.1087 参照) に 4 点内挿細分割曲線と、比較のために従来の B-spline 型単位 4 元数曲線 (階数 3) の例を示す。

3.4.4 計算時間の比較

表 1 に Chaikin と B-spline, 4 点内挿細分割と 3 種類の次数の B-spline 型単位 4 元数曲線による接線ベクトルの計算時間を示す。計算時間の比較に用いた曲線の制御接線 (c-tan.) の数はすべての曲線で 5 個と等しくさせたが、細分割曲線では細分割の深さに依存して計算される接線ベクトル (tan.) の個数が規定されてしまい、すべての曲線に対して一致させることはできない。B-spline 型単位 4 元数曲線ではパラメータの分割数を自由に変更できるので、任意の数の接線ベクトルを計算することができる。表 1 に示した例ではパラメータを 256 に分割し 257 個の接線ベクトルを計算している。計算時間の単位はミリ秒である。

Chaikin のアルゴリズムは、接線ベクトルの計算数が他の 1.5 倍程度であるにもかかわらず非常に計算時間が短い。これは 3.4.1 項で述べたように、式 (31) と (32) が 2 つのベクトルの和とその単位ベクトル化の繰返して計算でき、高速に処理可能だからである。同様な理由により、B-spline 細分割も高速に計算できるが、Chaikin のアルゴリズムに比較すると複雑であり、より時間がかかっている。4 点内挿細分割であっても、B-spline 型単位 4 元数曲線に比べると 4 倍から 5 倍高速に計算することができる。

4. ガウス球上の細分割曲面とその応用

ここでは、ストリームラインにガウス球上の細分割

曲面を利用することを考え、特に Kobbelt 型¹¹⁾ の細分割曲面を用いる。Kobbelt 型細分割は 4 点内挿細分割の曲面への拡張であり、Chaikin のアルゴリズムや B-spline 型の細分割に比較して処理速度の点では不利であるが、設計者が指定した接線方向を内挿するように分割がなされ、接線という位置に比べて直感的でない量の指定においては、接線方向が内挿されることが設計を行ううえで利点となる。細分割曲面としては、Doo-Sabin や Catmull-Clark がよく知られているが、非四辺形領域に対する接線ベクトルの内挿はパラメータ方向の選択に問題があり、それらの細分割形式を接線ベクトルの混ぜ合わせに用いることは未解決問題である。

4.1 Kobbelt 型細分割

Kobbelt 型細分割は 4 点内挿細分割の曲面への単純な拡張である。4 辺形メッシュは局所的に $s_{i,k} = \{p_{i,k}, p_{i+1,k}, p_{i+1,k+1}, p_{i,k+1}\}$ の順に定義される。細分かされたネットの点 $p_{i,k}^{j+1}$ は 3 つのグループに分類される。頂点 (vertex-point) $p_{2i,2k}^{j+1} = p_{i,k}^j$ は内挿の条件により固定される。稜線点 (edge-point) $p_{2i+1,2k}^{j+1}$ と $p_{2i,2k+1}^{j+1}$ はグリッドの方向に合わせて 4 点内挿細分割の規則を適用して算出する：

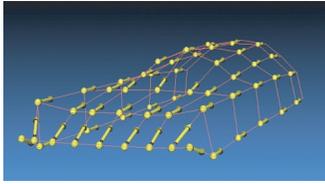
$$p_{2i+1,2k}^{j+1} = \frac{8+\omega}{16} (p_{i,k}^j + p_{i+1,k}^j) - \frac{\omega}{16} (p_{i-1,k}^j + p_{i+2,k}^j). \quad (42)$$

面点 (face-point) $p_{2i+1,2k+1}^{j+1}$ は連続する稜線点 $p_{2i+1,2k-2}^j, \dots, p_{2i+1,2k+4}^j$, あるいは $p_{2i-2,2k+1}^j, \dots, p_{2i+4,2k+1}^j$ に 4 点内挿細分割の規則を適用することにより算出する。

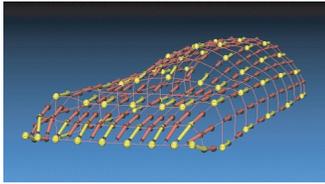
4 点内挿細分割と同様にガウス球上に拡張できるが、 s と t の 2 つのパラメータ方向に関して規則の適用順序による選択が残る。適用順序の違いによる影響は明らかでなく、その計算量は同じなので本研究で開発したシステムではユーザが選択できるようにしている。

4.2 ターゲット曲面例

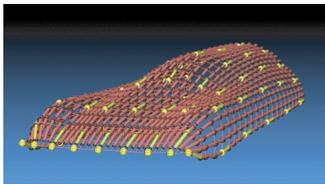
4 点内挿細分割曲線を曲面に拡張した Kobbelt 型のガウス球上の細分割曲面を用いて、曲面モデリングシステムを開発した。このモデリングシステムにより生成した曲面の例を図 6, 図 7 に示す。これらの例では式 (1) において $s_0 = 0$ として初期曲線 $S(s_0, t)$ を与えている。曲面の初期形状は平面であり、初期曲線は 4 つの接線ベクトルで、また $s(s, t)$ は 4×4 の接線ベクトルを混ぜ合わせるにより定義される。初期形状からユーザは接線ベクトルの方向を変更することで曲面を設計していく。接線ベクトルの方向は接線ベ



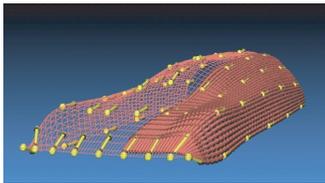
(a) オリジナルメッシュ (細分割深さ = 0)



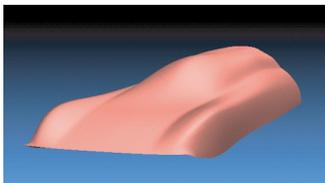
(b) ターゲット曲面 (細分割深さ = 1)



(c) ターゲット曲面 (細分割深さ = 2)



(d) ターゲット曲面 (細分割深さ = 3)

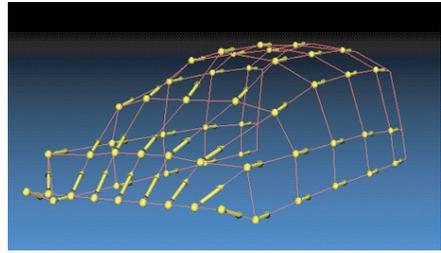


(e) ターゲット曲面 (細分割深さ = 3)

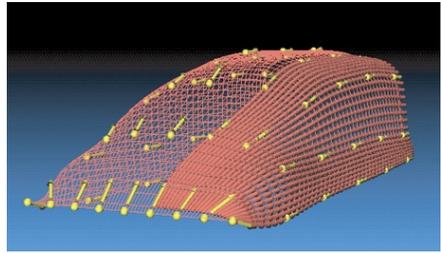
図 6 カーモデル No.1

Fig. 6 Car model No.1.

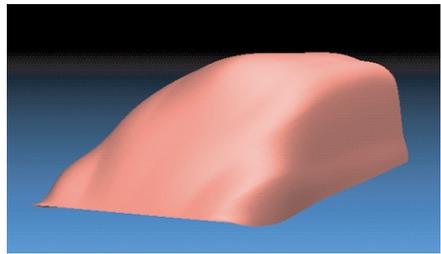
クトルの方向指定専用の GUI²⁾で行うとともに、数値による指定もできる。また、初期曲線を含めて接線ベクトルの本数を増減することができるように、初期曲線の長さ、パラメータ値 t に対する積分長も t についての多項式として変更できる。表示における細分割の深さはユーザが適宜指定可能であり、深さを適切に制御することにより会話を保ちながら曲面を設計することができる。



(a) オリジナルメッシュ (細分割深さ = 0)



(b) ターゲット曲面 (細分割深さ = 3)



(c) ターゲット曲面 (細分割深さ = 3)

図 7 カーモデル No.2

Fig. 7 Car model No.2.

5. おわりに

本研究では、接線の線形結合を行う演算を定義しガウス球上で定義される細分割曲線・曲面を提案した。さらにそれらの細分割曲面を用いて、ターゲット曲面の接線方向の混ぜ合わせ計算と表示処理の高速化を行い、これらの手法に基づく曲面モデリングシステムを開発した。このシステムにより、細分割の深さを調整しながらモデリングを行うことで会話を維持し、ストリームラインモデリングを用いた曲面設計が、接線方向の入力や曲面のレンダリング等を含めて実時間で行えることを確認した。

細分割曲面の大きな利点の 1 つは、任意の位相に対する曲面の生成であり、Doo-Sabin, Catmull-Clark 等の任意の位相に対する細分割手法を接線ベクトルの混ぜ合わせに用いることは今後の研究課題である。

参 考 文 献

- 1) 三浦憲二郎：単位 4 元数積分曲線，情報処理学会論文誌，Vol.38, No.11, pp.2227-2236 (1997).
- 2) Miura, K.T., Sakiyama, N., Kaneko, T. and Kubo, T.: Streamline Modeling: Designing Free-Form Surfaces by Blending Tangent Vectors, *6th IFIP WG5.2 International Workshop on Geometric Modeling: Fundamentals and Applications (GEO-6)*, pp.168-179 (1998).
- 3) 三浦憲二郎：単位 4 元数積分曲面，情報処理学会論文誌，Vol.41, No.3, pp.722-732 (2000).
- 4) 神野元紀，北野宏哉，三浦憲二郎，金子 透：単位 4 元数積分曲面による回転体の表現とその変形，情報処理学会論文誌，Vol.41, No.3, pp.522-530 (2000).
- 5) Miura, K.T.: Unit Quaternion Integral Curve: A New Type of Fair Free-Form Curves, *Computer Aided Geometric Design*, Vol.17, pp.39-58 (2000).
- 6) Farin, G.: *Curves and Surfaces for CAGD*, 4th Ed., p.120, Academic Press (1997).
- 7) Kim, M.-J., Kim, M.-S. and Shin, S.Y.: A General Construction Scheme for Unit Quaternion Curves with Simple High Order Derivatives, *Computer Graphics (Proc. SIGGRAPH '95)*, pp.369-376 (1995).
- 8) Chaikin, G.: An Algorithm for High-Speed Curve Generation, *Computer Graphics and Image Processing*, No.3, pp.346-349 (1974).
- 9) Schroder, P. and Zorin, D.: Subdivision for Modeling and Animation, *SIGGRAPH'98 course note* (1998).
- 10) Dyn, N. and Levin, D.: Interpolating Subdivision Schemes for the Generation of Curves and Surfaces, *Multivar. Approx. and Interp.*, Hausmann, W. and Jetter, K. (Eds), Birkhauser Verlag, Basel (1990).
- 11) Kobbelt, L.: Interpolatory Subdivision on Open Quadrilateral Nets with Arbitrary Topology, *Computer Graphics Forum 15 (Eurographics '96)*, pp.409-420 (1996).

(平成 12 年 10 月 2 日受付)

(平成 13 年 3 月 9 日採録)



三浦憲二郎(正会員)

昭和 34 年生。昭和 57 年東京大学工学部精密機械工学科卒業。昭和 59 年同大学院修士課程修了。同年，キャノン(株)入社。機械系 CAD/CAM システムの開発に従事。平成 3 年コーネル大学機械工学科博士課程修了(Ph.D.)。平成 5 年会津大学コンピュータ理工学部コンピュータソフトウェア学科助教授。平成 9 年静岡大学工学部機械工学科助教授。曲線・曲面の設計，CAD/CAM，要素自動分割，マイクロマシン等に興味を持つ。ACM，IEEE 各会員。