

NURBS曲線・曲面を対象とした 幾何形状を保存する電子透かし埋め込み手法

大淵 竜太郎¹, 増田 宏², 青野 雅樹³

ohbuchi@acm.org, masuda@race.u-tokyo.ac.jp, aono@acm.org

¹ 山梨大学工学部 コンピュータ・メディア工学科, 山梨県甲府市武田 4-3-11

² 東京大学 人工物工学研究センター, 東京都目黒区駒場 4-6-1

³ 日本アイ・ピー・エム 東京基礎研究所, 神奈川県大和市下鶴間 1623-14

要旨

3次元ポリゴンメッシュやその属性を対象とした従来の情報埋め込みアルゴリズムは, そのままでは(形状)CADモデルには適用できない事が多い. これには, CADモデルの多くは主たる形状定義プリミティブとしてポリゴンではなくパラメータ曲線や局面を用いていること, CADモデルは電子透かし埋め込みに伴うトポロジや幾何形状の変更を許さない場合が圧倒的に多いこと, の2つの理由がある.

本論文は, non-uniform rational B-spline (NURBS) 曲線および曲面を対象とした新たなデータ埋め込みアルゴリズムを提案する. このアルゴリズムは, 再パラメータ化に用いる有理線形関数の持つ自由度を使って情報を埋め込む. 再パラメータ化は NURBS 曲面・曲線の幾何形状を厳密に保存し, また, 有理線形関数による再パラメータ化は NURBS の次数や節点数を変えないためモデルのデータ量も保存する.

本論文ではさらに, NURBS に限定しない種々のパラメータ曲線・曲面を対象とした情報埋め込み手法の概略を, 幾何形状の保存とデータ量の保存という2つの要件で分類・整理して列挙した.

キーワード: CAD, 情報セキュリティ, 電子透かし, 再パラメータ化.

1. 初めに

データ埋め込み, または電子透かしと呼ばれる技術は, watermark(透かし)と呼ばれる何らかの構造体を, その存在が埋め込み対象となるコンテンツ本来の目的(例えば人による表示・鑑賞)を阻害しないように付加する. 埋め込まれた透かしは, 説明の追加, 改ざんの検出, あるいは正規の購入者の認証など, そのコンテンツを何らかの形で管理する目的で用いられる.

これまでのデータ埋め込みの研究の多くは「古典的」マルチメディアデータ型, 例えば文字文書, 静止画像, 動画画像, および音声データに対する埋め込みを中心としていた. しかし最近, 3次元(3D)データがその重要性を増すにつれ [VRML, MPEG4], 3Dモデルに対する埋め込みアルゴリズムがいくつか発表された.

我々は 3Dモデルの形状およびその属性を対象として情報を埋め込む手法を何種か発表してきた [Ohbuchi97, Ohbuchi98a, Ohbuchi98b]. 我々のアルゴリズムは 3Dポリゴンメッシュによって定義された形状を主な埋め込みの対象としている. アルゴリズムのうち幾つかは, 頂点座標から誘導

されたあるクラスの幾何変換に対する不変量を利用して透かしを埋め込んでおり, 3Dモデルが日常的にさらされる幾何変換, 例えばアフィン変換によって透かしが壊れないようになっている. 我々はこの他, 3Dモデルのトポロジ, つまり頂点の結合関係を変更して埋め込む透かしや, 3Dモデルの形状に付随する属性, 例えば頂点に付随するテクスチャ座標を変更してデータを埋め込む手法 [Ohbuchi98b] も提案した.

Kanai らは 3角メッシュをウェーブレット変換, その変換領域のデータを操作して情報を埋め込む手法を提案した [Kanai98]. 彼らの透かしはアフィン変換に耐え, かつ頂点座標に重畳されたランダムノイズに対してもある程度の頑強性を有した. Praun らの手法 [Praun99] も Kanai らの手法に近く, ウェーブレット分解を用いて変換領域で透かしの埋め込みを行った. 2次元の静止画像に対して初めて適用された fragile watermark(脆い透かし, 後述)を 3Dモデルに適用した例が [Yeo99] に述べられている. Benedens [Benedens99] は 3Dモデルの形状から求められた法線ベクタの集合を対象として, あるクラスの座標変換に対し頑強性を持つ透かしの埋め込み手法を述べた. また, 形状そのものを対象としてはいないが, Hartung ら [Hartung98] は MPEG-4 の facial animation parameters(顔アニメーションパラメータ; FAP) 列に対し情報を埋め込んだ. 興味深いのは, 顔アニメーションを 2次元動画画像にレンダリングした結果から透かしを取り出すことにも成功した点である.

これまでの 3D形状を対象とするデータ埋め込みアルゴリズムは, そのままでは, 形状 CAD データには適用できない場合が多い. これには 2つの理由がある. 第一に, 大多数の CADモデルはその形状を定義するのにポリゴンメッシュを用いず, パラメータ形式の曲線や曲面, 例えば Bezier や non-uniform rational B-spline (NURBS)形式の曲線や曲面を用いる. 従って, ポリゴンメッシュ用に開発された透かしのアルゴリズムをそのまま適用することはできない. 第二に, 殆どの CADモデルはその幾何形状やトポロジの変更を加えられると使い物にならなくなるため, 座標値や頂点のトポロジを変更するこれまでの電子透かしアプローチはそのままでは使用できない.

本論文では, パラメータ形式の曲線や曲面を対象とした情報埋め込みアルゴリズムを提案する. 本論文の貢献は以下のようにまとめることができる.

- NURBS 曲線・曲面を対象に, rational linear reparameterization (有理線形再パラメータ化)を用いて情報を埋め込む具体的手法を提案した. この手法は, 透かし埋め込みの前後で NURBS 曲線・曲面の形状を厳密に保存し, さらにモデルのデータ量も保存する.

¹ 本論文で紹介する研究は, 第一筆者の大淵が IBM 東京基礎研究所の主任研究員であった時期に行った.

- NURBS に限らず、各種のパラメタ形式の曲線や曲面にデータを埋め込む手法の概略を、これらのアプローチ、(1) 形状を保存するか、(2) データ量を保存するか、の2つの性質によって分類し、列挙した。

以下、まず本節の残りの部分でデータ埋め込みの概念一般について簡単に紹介する。第2節では、NURBS 曲線・曲面を対象に、形状とデータ量を変えずに透かしを埋め込むアルゴリズムを紹介し、第3節ではこのアルゴリズムの実装と実験結果について述べる。第4節では、NURBS に限定しない種々のパラメタ曲線・曲面を対象とした電子透かしのアプローチを列挙する。最後に、第5節でまとめを述べ、これからの研究方向についてコメントする。

1.1. 情報埋め込み

本論分では、透かしをコンテンツに追加することを(データ)埋め込み(embedding)と呼び、また埋め込まれた情報を目的に応じて取り出すことを取り出し(extraction)と呼ぶ。透かしが埋め込まれる対象物を被覆-<datatype>、透かしの入った対象を隠号化-<datatype>、そして埋め込まれた情報を埋め込み-<datatype>と呼ぶ [Pfitzman96]。ここで "<datatype>" はデータ型によって変わり、テキスト、画像、あるいは NURBS 曲線・曲面などとなる。

電子透はその可視性 (visibility)、より一般的には可知性 (perceptibility) で分類される。可知な透かしは、コンテンツの通常の使用環境下で人にとって可知であるが、不可知な透かしは何らかの機械的な処理を経てはじめて可知となる。不可知な透かしが一般的であるが、可知な透かしもその性質を積極的に使い、コンテンツの無許可の再配布を牽制するなどの使い道がある。

画像などを対象とした透かしの可知性による分類では被覆データの観測者が人であると仮定してきた。しかし、3D CAD モデルが被覆データの場合、3D モデル自体やその透かしの観測は間接的である。3D モデルが人にとって可知になるのは、レンダリングおよび表示の処理を経た後、あるいはそのモデルで定義された機械部品などが実際に製造された後であり、3D モデルの可知性の定義は2次元画像等の場合とは異なる。

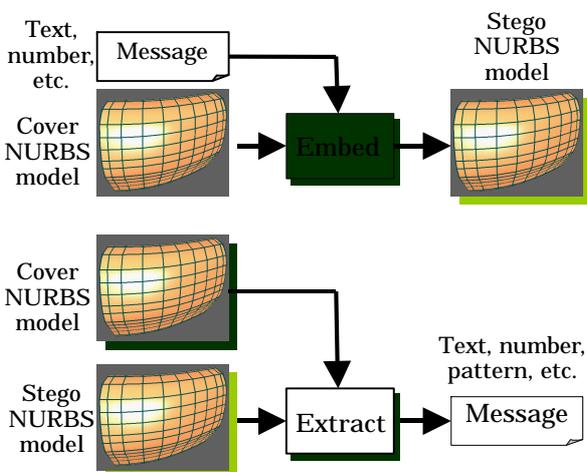


図1. NURBS 曲面・曲線を対象とする秘密透かし手法のデータの流れ。取り出しには透かしの入っていない被覆 NURBS モデルが必要である。

電子透かしはまた頑強な透かし (robust watermark) と脆弱な透かし (fragile watermark) に分類できる。前者はコンテンツに加えられる変更が意図的 (intentional) か非意図的 (unintentional) かにかかわらず耐えて保存される必要がある。後者は、あるコンテンツに対する意図的な変更の全て、および一部の非意図的な変更によって改変されることにより、そのコンテンツの改ざんや変更、あるいは改ざん部分のを検出しようとするものである。ここで非意図的な変更とは、コンテンツの通常の使用において予期されるもの、例えば画像における JPEG 圧縮・伸長などを指し、これに対し意図的な変更とは、透かしを変更又は破壊するなどの目的で意図的に加えられるものを指す。

別な分類として、透かしの取り出しに際して透かしの入った隠号データに加え、オリジナル(透かし埋め込み前)の被覆データを必要とするとき、その透かしは秘密透かし (private watermarking) と呼び、オリジナルがいらないとき公開透かし (public watermarking) と呼ぶ。例えば Hartung ら [Hartung98] の手法は秘密透かしである。

また、暗号の考え方にを使い、鍵を使って埋め込み情報を保護することもできる。例えば第三者による情報取り出しを避けるため、埋め込み時に(擬似)乱数系列を用いて透かしを埋め込んだ場所をかく乱すること、あるいは透かしのメッセージ自体をかく乱するなどすることもある。この乱数系列生成の為に鍵を隠号鍵 (stego-key) と呼ぶ。隠号鍵と乱数系列の関係は、秘密鍵暗号式でも公開鍵暗号式でも良い [Menezes96]。乱数列によるかく乱にはまた、[Hartung98] のように拡散帯域通信の考え方を使って埋め込みによる変更を分散し、妨害耐性を高めたりする使い方もある。

気をつけねばならないのは、その他の各種セキュリティ技術と同様に、電子透かし技術単体では、著作権保護などの現実の応用で効果を上げることはできない点である。実際に何が重要なのかはそれぞれの応用で異なるが、透かしと組み合わせる電子署名や、透かしを入れたデータを預託する為の信用できる預託機関などのインフラストラクチャ、そしてこれらインフラストラクチャの社会的・法的な認知など、満たされねばならない要件は数多い。

2. NURBS への形状不変な情報埋め込み

本節では、NURBS 曲面・曲線を対象とし、幾何形状を厳密に保存し、かつデータ量も変化しない情報埋め込み手法を述べる。本手法は秘密透かしで、従って取り出しにはオリジナルの被覆 3D モデルと透かしの入った隠号 3D モデルの両方が必要である(図1参照)。

形状を保存する本透かし手法の基本的アイデアは、NURBS 曲面・曲線はその幾何形状を全く変えることなく再パラメタ化することができる、という事実に基づいている。以下、本節ではまず NURBS 曲線についてアルゴリズムを説明し、後にテンサー積として作られた NURBS 曲面に拡張する。

幾何形状を変えない性質(形状保存性)は CAD の用途の多くで要求される。例えば、多少とも幾何形状が変形したモデルを使った Constructive Solid Geometry 演算は失敗するであろう。また、データ量を変えない性質(データ量保存性)も通信や蓄積のコスト削減に重要である。

NURBS 曲面・曲線モデルのデータ量は主に control point (制御点) と weight (重み) の数と knot vector (節点ベ

クタ)の要素数で決まが、本節で述べる手法は制御点と重みの数や節点ベクタ要素の数を変えないため、モデルのデータ量に変化は無い。ただ、本論分の手法でも、「データ圧縮をかけた後のビット数」は埋め込み前後で変わり得る。

2.1. NURBS 曲線

本節ではまず簡単に NURBS 曲線を定義する。NURBS そのほかのパラメータ曲線・曲面について詳しくは Farin [Farin97] や Piegl と Tiller [Piegl97] などを参考にしてほしい。本論分では Piegl と Tiller の本の定式化を用いた。

p 次の NURBS 曲線 $C(u)$ は、スケール値 u が区間 $[a, b]$ で変化するにつれ、ある点が 3 次元空間をたどる軌跡を定義する。

$$C(u) = \frac{\sum_{i=0}^n N_{i,p}(u) w_i \mathbf{P}_i}{\sum_{i=0}^n N_{i,p}(u) w_i} \quad u \in [a, b], \quad (1)$$

ここで制御点 $\{\mathbf{P}_i\}$ の集合は control polygon (制御ポリゴン) をなし、また $\{w_i\}$ は重みである。重み w_i の増加は曲線を制御点 \mathbf{P}_i に引き寄せる効果がある。また $N_{i,p}(u)$ は、次数 p (階数 $p+1$) の i 番目の B-spline 基底関数で、再帰的に次のように定義される。

$$N_{i,0}(u) = \begin{cases} 1 & \text{if } u_i \leq u < u_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u). \quad (2)$$

Non-periodic (非周期的) かつ non-uniform (非一様) な knot vector (節点ベクタ) は区間 $[a, b]$ 内で非減少 (増加するか値が変わらない) な実数値の列であり、以下のように定義される。

$$U = \underbrace{\{a, \dots, a, u_{p+1}, \dots, u_{m-p-1}, b, \dots, b\}}_{p+1} \quad (3)$$

ここで $a \leq u_i \leq u_{i+1} \leq b$ and $i = 0, \dots, m-1$ である。

2.2. NURBS 曲線の再パラメータ化

パラメータの区間 $u \in [a, b]$ で定義された NURBS 曲線 $C(u) = \{x(u), y(u), z(u)\}$ を関数 $u = f(s)$ で reparameterize (再パラメータ化) すると、この曲線は、もとのパラメータ u の代わりに新たなパラメータ s の関数として計算される。ここで、曲線上の点の軌跡が同一の点を 2 回以上通らないよう、 $f(s)$ は区間 $s \in [c, d]$ 内で増加関数 (つまり、 $f'(s) > 0, \forall s \in [c, d]$) と限定する。再パラメータ化についての詳細は [Tiller83] や [Tiller97] を参照してほしい。重要なのは、再パラメータ化は NURBS 曲線・曲面の形状を保存するてんである。

再パラメータ化関数 $f(s)$ は数多く存在するが、任意の再パラメータ化関数で再パラメータ化した結果は NURBS 曲線とは限らない。例えば、次数 2 以上の多項式で再パラメータ化した結果は NURBS 曲線であるが、その次数は元の曲線 $C(s)$ より高く、従ってより多くの制御点を必要とし、モデルのデータ量が増える。

本論文の手法では有理線形式で再パラメータ化する。こうすると再パラメータ化の前後で NURBS 曲線の次数に変化が

無く、燃って制御点数の増加などのデータ量の増加を伴う変化がない。有理線形 (有理 1 次, Moius, などとも呼ばれる) 式による再パラメータ化は Lee と Lucian [Lee91] が調べており、本論文の手法はその結果を用いている。以下 Lee らの結果を要約する。

有理線形関数 $g(u)$ は以下のように定義される。

$$s = g(u) = \frac{au + b}{gu + d} \quad (1)$$

$$u = f(s) = \frac{-ds + b}{gs - a} \quad s \in [c, d] \quad (2)$$

ここで $f(s)$ は $g(u)$ の逆関数である。ここで

$$\mathbf{m}(u) = gu + d \quad \mathbf{l}(s) = gs - a \quad (3)$$

とおく。 $g(u)$ と $f(s)$ の振る舞いが良いためには

$$ad - gb > 0, \quad \mathbf{m}(u) \neq 0 \quad \text{for all } u \in [a, b], \quad \text{および} \quad \mathbf{l}(s) \neq 0 \quad \text{for all } s \in [c, d] \quad (4)$$

が必要である。NURBS 曲線 $C(u)$ が与えられたとき、これを $g(u)$ で再パラメータ化した $C(s)$ は以下のようにして求まる:

- 制御点 $\{\mathbf{P}_i\}$ は不変。
- 新しい knot (節点) ベクタは元の節点ベクタの個々の節点を $g(u)$ で写像したもの、つまり $s_i = g(u_i)$ 。
- 制御点の新たな重み $\{\bar{w}_i\}$ は (5) 式のいずれか一方で計算する;

$$\bar{w}_i = w_i \prod_{j=1}^p I(s_{i+j}) \quad \text{または} \quad \bar{w}_i = \frac{w_i}{\prod_{j=1}^p \mathbf{m}(u_{i+j})} \quad (5)$$

ここで s_{i+j} と u_{i+j} はそれぞれ新・旧の節点である。

2.3. 埋め込みプリミティブとしての NURBS 曲線

本論分ではデータ埋め込みのためコンテンツに変更を加える最小単位を embedding primitive (埋め込み単位) と呼ぶ。従って有理線形再パラメータ化を行って情報を埋め込んだ NURBS 曲線は埋め込みプリミティブである。埋め込みプリミティブ 1 つあたりの埋め込み情報量は一般に小さいが、複数の埋め込みプリミティブを組み合わせると有用な量の情報を埋め込むことができる。

自由度 3 の有理線形再パラメータ化関数 $g(u)$ に情報を埋め込むには、その係数 a, b, g , および d を操作すれば良い。この関数の自由度は、係数が 4 つあるにもかかわらず 3 であることは以下の変形から明らかである:

$$s = g(u) = \frac{au + b}{gu + d} = \frac{\frac{a}{g}u + \frac{b}{g}}{u + \frac{d}{g}} = \frac{k_1 u + k_2}{u + k_3}, \quad (10)$$

ここで $k_1 = a/g$, $k_2 = b/g$ および $k_3 = d/g$ と置いた。

これら 3 つの係数 k_1, k_2 , および k_3 は、関数 $g(u)$ が通る 3 点 (u_1, s_1) , (u_2, s_2) , and (u_3, s_3) を決めることで決定される (図 2)。この 3 点を (4) 式に代入して連立式として $k_1,$

k_2 , および k_3 について解くと以下ようになる:

$$k_1 = \frac{(u_1 s_1 - u_2 s_2)(s_1 - s_3) - (u_1 s_1 - u_3 s_3)(s_1 - s_2)}{(u_1 - u_2)(s_1 - s_3) - (u_1 - u_3)(s_1 - s_2)}$$

$$k_2 = u_1 s_1 + k_3 s_1 - k_1 u_1 \quad (11)$$

$$k_3 = \frac{(u_1 s_1 - u_3 s_3)(u_1 - u_2) - (u_1 s_1 - u_2 s_2)(u_1 - u_3)}{(u_1 - u_3)(s_1 - s_2) - (u_1 - u_2)(s_1 - s_3)}$$

式(1)の持つ 3 自由度全てを操作して情報を符号化することも可能だが,ここでは 2 端点を $u_1 = s_1 = a$ および $u_3 = s_3 = b$ で固定し,パラメタの区間が再パラメタ化の前後で同じになるようにした.情報は残った 1 自由度を使い, $D = s_2 - u_2$ を変更して符号化する.ここで,偏移値 D はそ

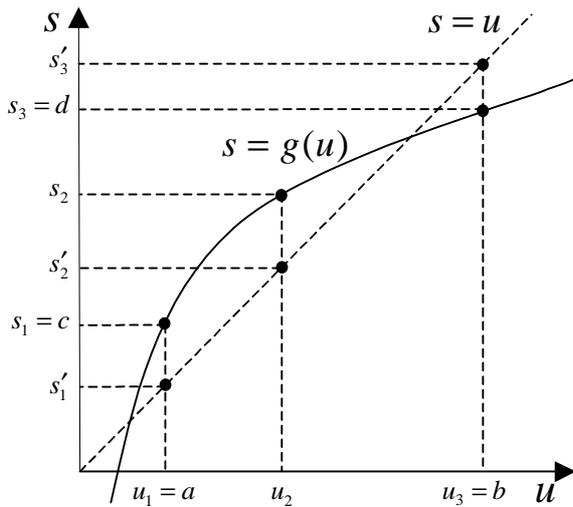


図 2. 再パラメタ化に使用する自由度 3 の有理線形関数を決定する.

の値が大きいくほど関数 $s = g(u)$ は直線 $s = u$ から離れる.

偏移値 D の大きさには注意が必要である. D が大きすぎるとパラメタ化が偏り,「悪い」パラメタ化とされる.(パラメタ化が「良い」とときには,パラメタ空間上で等間隔の点が実空間の曲線上でもほぼ等間隔に写像される事を言う.) D の値が小さければ,「良い」パラメタ化をもつ曲線は,ほぼ同様に「良い」パラメタ化を持つ曲線に写される.もちろん D が小さすぎると計算誤差などによる影響を受けやすいので,適度な値を選ぶ必要がある.我々のプロトタイプでは D のとり得る範囲をユーザが指定する.

具体的には偏移値 D を直接に振幅変調して値を埋め込んだ. D のとり得る範囲 $[D_{\min}, D_{\max}]$ に L ビットのデータ d を符号化するとすると,偏移値 D の値は以下の式で求める:

$$D = \frac{(D_{\max} - D_{\min})(d + 0.5)}{2^L} + D_{\min} \quad (12)$$

この D の値により, NURBS 曲線の interior knot(内部節点)の位置を変更する.幾何形状を不変に保つため,節点の位置を変更した結果として節点の値および制御点の重みが変わるが,制御点座標値そのものはまったく変わらない.

埋め込み対象となる NURBS 曲線には少なくとも 1 つの内部節点が必要である.(内部節点を持たない NURBS 曲線は rational Bezier(有理 Bezier) 曲線である.)現在の実

装では,内部節点が m 個ある場合には,常に $i = \lfloor m/2 \rfloor$ で求まる i 番目の内部節点を埋め込みに用いた.しかし,どの節点を用いるかは,鍵を用いて作った乱数列などで変えることもできる.

情報の取り出しは,元々の曲線(被覆 NURBS 曲線)と再パラメタ化後の曲線(隠号 NURBS 曲線)の節点ベクタ値を比較し,埋め込み時と同じ節点でその差 $D = s_2 - u_2$ を求め,これから式(12)を用いてデータ d を取り出す.

実際には,乱数系列を用いて埋め込みメッセージをかく乱することでデータのもつ規則的なパターンを消すことが望ましい.また,複数の曲線・曲面に対して埋め込みをする場合,埋め込みをしない曲線・曲面も再パラメタ化を行い,情報の埋め込まれた曲線・曲面が容易に同定できないようにすることが望ましい.

2.4. NURBS 曲面への埋め込み

ここまでは簡単のため曲線で説明したが,このアルゴリズムは 2 つの NURBS 曲線のテンサー積として作られる NURBS 曲面に容易に拡張できる. NURBS 曲面が u と v でパラメタ化されていたとするなら,前節の手法をこれら 2 つのパラメタそれぞれに適用し,2 つの偏移値 D_u を D_v を使って曲面を再パラメタ化すれば良い. NURBS 曲線に L ビット埋め込めたとすると, NURBS 曲面には $2L$ ビットが埋め込める.

2.5. 埋め込みプリミティブの順序付け

埋め込みプリミティブ当たりで埋め込めるデータ量は少ない.しかし埋め込みプリミティブ間で順序をつけることにより,複数のプリミティブ全体で大きな容量のデータを埋め込むことが可能である. CAD モデルの場合は,ひとつのモデルを構成する曲面,曲線に識別番号がついており,これによってプリミティブの順序付けを実現できる.現実の CAD モデルは数百,数千を超える曲面,曲線(特にトリム曲線)からなっている.したがって CAD モデル全体としては種々の用途に十分な量のデータを埋め込むことが可能であろう.

埋め込みに際しては,プリミティブの順序とプリミティブに埋め込まれるメッセージの間の対応を,公開鍵暗号などを用いた擬似乱数列によってかく乱することが考えられ,これにより第三者による情報の取り出しなどを防ぐことができる.

2.6. 本埋め込み手法の特性と使い道

ここで述べた再パラメタ化による埋め込み手法は,形状保存性,データ量保存性という 2 つの大きな特長を持っているが,欠点はその除去が容易な事である.埋め込まれたデータを除去するには曲面・曲線に任意の再パラメタ化を加えれば良い.従って,本節で述べた手法は脆弱な透かしとして使い,認証や改ざんの検出,例えば「正規の認証コードが入っていない場合は何らかの予期せぬ変更が加えられた」のような使い方が主になるであろう.

3. 実験と結果

我々は第 2 節に述べた NURBS 曲線・曲面を対象とした情報埋め込みアルゴリズムを実装した.現在の実装では,曲線当たり L ビット,曲面当たり $2L$ ビットの情報を埋め込む.曲線・曲面の埋め込みプリミティブ間の順序は CAD システムの持つ順序番号を用いる事とした.プログラムは C++ で書

き，表示には OpenGL を用いた．

図 3.(a)-(b) は 3 次 NURBS 曲線において 2 種類の偏移値 D で再パラメタ化の結果を示す．折れ線は制御点からなる制御多角形であり，各制御点は丸で示されている．曲線には，区間 $[a, b]$ 上に等間隔に置かれた，両端を含めて 10 個のパラメータ値に対応する 10 個のマーカーを付けてある．再パラメタ化前は十字の，再パラメタ化後は四角（ひし形）のマーカーで，パラメタ化を示してある．比較的小さな偏移値 $D = 0.01$ では再パラメタ化の前後でパラメタ化の変化は小さい．大き目の偏移値，例えば $D = 0.5$ では明らかに偏ったパラメタ化が認められる．

図 3. (c)-(d) は $L = 8$ -bit の情報を埋め込まれた NURBS 曲線を示す．8-bit の情報は節点を表現する倍精度浮動小数点数（64 bit）に埋め込まれている．

図 4. と図 5. は NURBS 曲面を再パラメタ化して情報を埋め込んだ例である．図 5. (a) では 2 つのパラメータ u と v の偏移値として $D_u = 0.001$ および $D_v = 0.02$ を用いた．実線が再パラメタ化前，点線が再パラメタ化後のパラメタ化をあらわしている．図 5.(b) は ASCII 文字 "aZ" の 2 文字の文字コードを合計 16 bit のデータとして NURBS 曲面に埋め込んだ例である．偏移値 D の範囲は $[D_{\min}, D_{\max}] = [0.001, 0.01]$ と意図的に大きめに取り，再パラメタ化によるパラメタ化の変化が見られるようにしてある．ASCII では "a" のほうが "Z" より値が大きいため，こちらのほうがパラメタ化の変化も大きい．埋め込みビット数 $L = 8$ の場合には， D_{\max} をこれよりかなり小さくしても埋め込みの安定性に影響は無い．

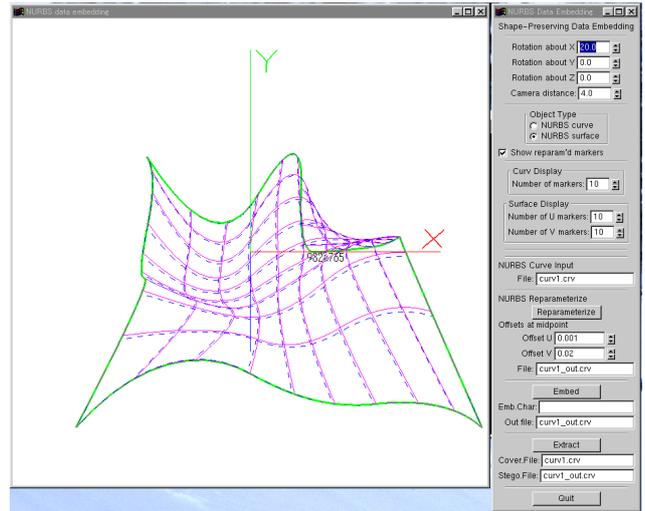
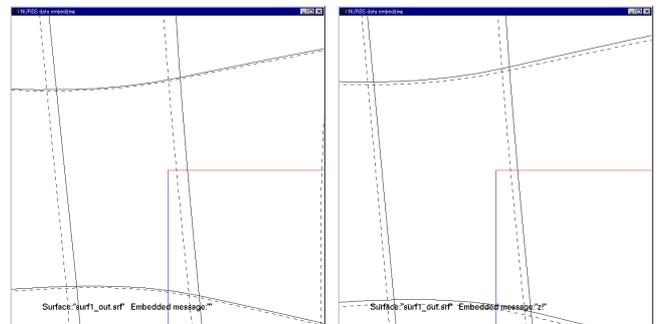
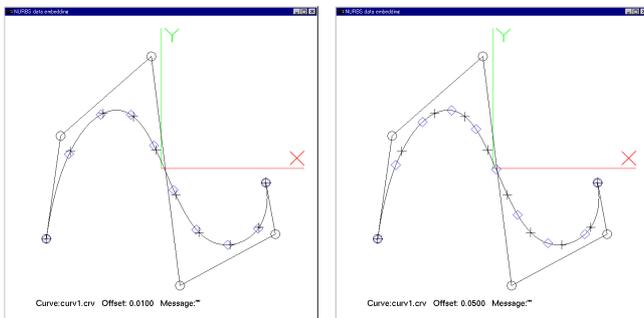


図 4. NURBS 曲面を $D_u = 0.001, D_v = 0.02$ の偏移値を用いて再パラメタ化した結果．実線が再パラメタ化前，点線が再パラメタ化後．



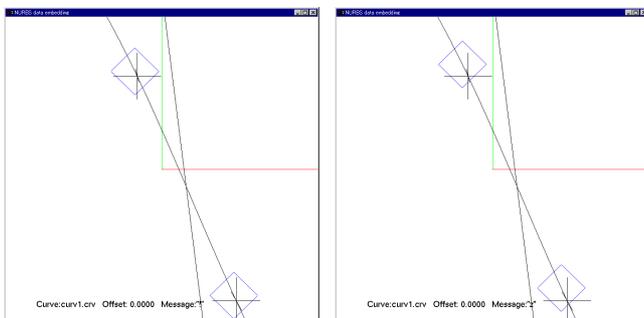
(a) $D_u = 0.01, D_v = 0.001$ で再パラメタ化した結果． (b) 2 文字，16bit のデータを埋め込んだ結果．

図 5. 上の図 4. と同一の曲面を拡大．(a) は曲面が $D_u = 0.01, D_v = 0.001$ で再パラメタ化された結果．(b) は文字列 "aZ" の文字コードを 2-byte のデータとして埋め込んだ結果．いずれも実線が再パラメタ化前，点線が再パラメタ化後．



(a) 偏移値 $D = 0.01$

(b) 偏移値 $D = 0.5$



(c) 文字 "!" を埋め込んだ結果（拡大図）．

(d) 文字 "z" を埋め込んだ結果（拡大図）．

図 3. NURBS 曲線を，図(a) は $D = 0.01$ ，(b) は $D = 0.5$ の偏移値で再パラメタ化した例．図 (c) と(d) は同じ曲線に文字 "!" と "z" のコードを埋め込んだ例．

4. 曲線・曲面に情報を埋め込むその他の手法．

第 2 章では NURBS 曲面・曲線を対象とし，形状不変かつデータ量不変に透かしを埋め込む方法を述べた．本節では，NURBS を含め，Bezier，有理 Bezier，B-spline などの種々のパラメトリック曲線・曲面を対象に情報を埋め込む手法についてその幾つかの概略を述べる．これら埋め込み手法は，形状保存性，すなわち幾何形状を厳密に保存するか，とデータ量保存性，すなわちモデルデータのサイズ保存するか，の二つの特徴で分類した．ただし，形状，データ量いずれの保存も不要な場合については特に言及しない．

4.1. 形状を保存し，データ量を保存しない手法

形状保存し，データ量を保存しない手法は，以下のようにして実現できる．

- Knot insertion (節点挿入)：複数のスパンからなるパラメータ曲線，例えば非有理 B-Spline や NURBS 曲線に節点を挿入する．その節点の値，節点の存在そのもの，

あるいは節点ベクタ中での新たな節点の(順序)位置, などを使得情報符号化し埋め込みむ。

- Degree elevation(次数上げ): 次数上げは制御点の追加を起こす。次数がどれだけ増えたか, どの区間に制御点を挿入したか, などに情報を埋め込むことができる。非有理な Bezier や B-spline に使用できる。
- 次数上げの伴う再パラメタ化: 有理 Bezier や NURBS では, 次数上げを伴う再パラメタ化により情報を埋め込むことができる。次数上げを伴う再パラメタ化は, 新たな制御点や節点を導入する。

上記のデータ埋め込み手法で導入された制御点や節点を, 形状を保存しつつ除去することはかなり困難であろう。従って, ここで述べたアプローチは著作権保護や秘密漏洩の追跡を目的とした頑強な埋め込みに向いていると思われる。その反面, データ量保存性を要求する利用目的には適当でない。

4.2. 形状を保存せず, データ量を保存する手法

形状の厳密な保存が必要ないがデータ量は保存したい場合の手法もいくつか考えられる。以下で述べる手法は, 第 4.2 節の手法ほどではないにしろ, 比較的高い頑強性を実現できる。

- 制御点座標値の変更: 制御ポリゴンをポリゴンメッシュとして扱えば, ポリゴンの頂点座標値を変えて埋め込む情報埋め込みアルゴリズムが使える。また, 頂点群には既に順序がついているため, その順序を用い, 制御点座標値を単なる数値列として扱ってその数値を変更して情報を埋め込める。この手法はほとんど全てのパラメータ曲線・曲面に適用できる。
- 節点ベクタ要素値の変更: 節点ベクタを単なる数値列として扱い, その値を変更して情報を埋め込む。NURBS や B-スプラインなど複数スパンからなり節点を持つパラメータ曲線・曲面に適用できる。

制御点座標値, 節点ベクタ要素値, いずれを変更する場合は, 値をそのまま変更しても良いし, 離散コサイン変換(DCT)やなどの変換を施した変換領域で変更しても良い。例えば, DCT 領域で埋め込む手法は, 我々が[増田 99]で提案した DCT を用いる lossy な曲面形状圧縮手法とうまく組み合わせることが可能だろう。

5. まとめと今後の課題

本論文では, 形状が NURBS 曲面・曲線で表現される CAD モデルを対象とし, 有理線形関数による再パラメタ化を用いることで, 曲面・曲線の形状を厳密に保存し, かつそのデータ量を保存しつつ情報を埋め込む手法を提案し, またその手法を実装して実験結果を示した。

本論文ではまた, NURBS に限らずパラメトリック曲線・曲面に対し情報を埋め込む複数の手法の概略を, 形状保存性とデータ量保存性を使得分類して列挙した。

NURBS 曲面・曲線を対象とする電子透かし手法の提案, 3次元形状を対象としその形状もデータ量も保存する電子透かし手法の提案, さらに各種のパラメータ曲線・曲面を対象とする電子透かし手法の提案は, 我々の知る限り, そのい

ずれをとって本論文が最初である。

今後, われわれは, 第 4 節で述べたような, 一般のパラメータ曲面・曲線を対象とした各種の情報埋め込み手法, 特に形状保存だが非データ量保存で頑強性の高い手法の実装を検討したい。さらに, これら複数の手法を使得, 実際の CAD データを対象とした埋め込み実験も行いたい。

参考文献

- [Benedens99] O. Benedens, Geometry-Based Watermarking of 3D Models, *IEEE CG&A*, pp. 46-55, January/February 1999.
- [Farin97] G. E. Farin, *Curves and Surfaces for Computer-Aided Geometric Design, A Practical Guide*, Fourth Edition, Academic Press, San Diego, CA, 1997.
- [Hartung98] F. Hartung, P. Eisert, and B. Girod, Digital Watermarking of MPEG-4 Facial Animation Parameters, *Computer and Graphics*, Vol. 22, No. 4, pp. 425-435, Elsevier, 1998.
- [VRML] ISO/IEC 14772-1 Virtual Reality Model Language (VRML).
- [MPEG4] ISO/IEC JTC1/SC29/WG11 *MPEG-4 Visual and MPEG 4 SNHC*.
- [Kanai98] S. Kanai, H. Date, and T. Kishinami, Digital Watermarking for 3D Polygons using Multiresolution Wavelet Decomposition, Proc. of the *Sixth IFIP WG 5.2 International Workshop on Geometric Modeling: Fundamentals and Applications (GEO-6)*, pp. 296-307, Tokyo, Japan, December 1998.
- [Lee91] E. T. Y. Lee, and M. L. Lucian, Mobius Reparameterizations of Rational B-splines, *Computer Aided Geometric Design*, Vol. 8, pp. 213-215, Elsevier, 1991.
- [増田99] 増田 宏, 大淵 竜太郎, 青野 雅樹, 周波数領域での曲面データの圧縮と転送, 情報処理学会論文誌, 第40巻第7号, pp.1188-1195, 1999年3月号。
- [Menezes96] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1996.
- [Ohbuchi97] R. Ohbuchi, H. Masuda, and M. Aono, Watermarking Three-Dimensional Polygonal Models, *Proceedings of the ACM Multimedia '97*, Seattle, Washington, USA, November 1997, pp. 261-272.
- [Ohbuchi98a] R. Ohbuchi, H. Masuda, and M. Aono, Watermarking Three-Dimensional Polygonal Models Through Geometric and Topological Modifications, pp. 551-560, *IEEE Journal on Selected Areas in Communications*, May 1998.
- [Ohbuchi98b] R. Ohbuchi, H. Masuda, and M. Aono, Geometrical and Non-geometrical Targets for Data Embedding in Three-Dimensional Polygonal Models, *Computer Communications*, Vol. 21, pp. 1344-1354, Elsevier (1998).
- [Piegl97] L. Piegl, W. Tiller, *The NURBS Book*, 2nd Edition, Springer, Berlin, 1997.
- [Pfitzmann96] B. Pfitzmann, Information Hiding Terminology, in R. Anderson, Ed., *Lecture Notes in Computer Science* No. 1174, pp. 347-350, Springer, 1996.
- [Praun99] Emil Praun, Hugues Hoppe, Adam Finkelstein, Robust Mesh Watermarking, MSR-TR-99-05, Microsoft Research, 1999.
- [Tiller83] W. Tiller, Rational B-splines for curve and surface representation, *IEEE Computer Graphics and Applications*, Vol. 3, No.6, pp. 61-69, 1983.
- [Yeo99] B-L. Yeo and M. M. Yeung, Watermarking 3D Objects for Verification, *IEEE CG&A*, pp. 36-45, January/February

1999.