境界曲線の内挿と曲面差分を用いたNURBS曲面データ圧縮

古川 慈之　　**増田 宏**
東京大学大学院工学系研究科　**東京大学人工物工学研究センター**

1　はじめに

3次元形状データを活用する場の多様化とネットワーク・インフラの急速な整備により，3次元形状モデルを対象としたネットワーク経由のデータ交換は今後増大していくことが予想される．しかし，このようなコンテンツのデータ交換においては，標準化の問題がある一方，転送データ量の増加による転送時間の増大という問題がある．この問題は，特に大規模なデータを扱うシステム間でのデータ交換において顕著である．

この問題を解決するためには3次元形状データの圧縮手法が必要である．3次元形状データの圧縮については多数の研究がなされてきたが，ポリゴンメッシュデータの圧縮が主流であり，曲面が多く用いられるモデルを対象としたものが少ない．CADやモデラで生成された3Dデジタルコンテンツを有効活用するためには，曲面データを対象に含め，利用目的に応じたデータ量の削減や精度の調整が可能な圧縮転送方式を考える必要がある．

本研究では，3次元形状データの有効活用を念頭に置いた形状データ圧縮手法の基礎的研究として，NURBS曲面モデルを対象としたデータ圧縮手法を提案する．

2　形状データ圧縮

2.1　関連する従来研究

形状データ圧縮に関する研究として，ポリゴンメッシュモデルと曲面モデルを対象とした主な研究について概観する．

ポリゴンメッシュモデルを対象とした研究としては，Deering[1]の研究がよく知られている．Deeringは3次元空間のベクトルを極座標表現に置き換え，二つの角度成分を離散的な値に畳み込むことでデータ量の削減を実現した．また，三角形メッシュによる多面体表現において，頂点インデックスを利用して重複する頂点座標を省略することでデータ量を削減した．

Taubinら[2]は，三角形メッシュデータから三角形ストリップを切り出して接続情報を効率的に圧縮する研究を行った．また，多角形や非多様体を分割して三角形メッシュの圧縮に帰着させる方法についても示している．三角形メッシュの接続情報を圧縮する研究としては，Gumholdら[3]，Rossignac[4]による手法なども報告されている．また，増田ら[5]は，オイラー操作列の符号化と組み合わせて，穴や空洞，セルフループを含む任意のCADデータの位相構造を三角形メッシュの圧縮に帰着させる方法を提案している．

Hoppe[6]は、形状簡略化をベースにしたプログレッシブ転送手法を提案した．この方法では，粗い情報から詳細な情報へと段階的に形状データを送信することができる．プログレッシブ転送には，他に，Liら[7]，Taubinら[8]，Pajarolaら[9]，Khodakovskyら[10]，Balanら[11]などの手法も提案されている．

このように，現在までにメッシュモデルに対する圧縮転送手法は数多く研究されているが，一方で，曲面モデルに対する圧縮手法は研究例が少ない．

Devoreら[12]は陰関数曲面に対し，ウェーブレット変換を用いた圧縮手法を提案した．また，増田ら[13]は，パラメトリック曲面を対象にして，離散コサイン変換を用いた圧縮手法を提案している．これらの手法は、複数の曲面からなるモデルを圧縮する場合，別々に量子化された曲面の境界部で不連続となり，品質の劣化が目立つという問題が生じる．

また，脇田ら[14]は曲面データを圧縮が容易な独自の曲面表現に変換することで高い圧縮率を実現している．この方法はデータ閲覧には適しているが，独自形式の曲面表現に近似しているために，元の表現を完全に再現することは困難である．そのため，形状データの利用分野が限定されてしまうという問題がある．

2.2　解決すべき問題点と研究の範囲

3次元形状データの有効活用を目的として，曲面モデル圧縮手法の構築に関して現状で解決すべき問題点を次のように整理する．

1. データ量：精度の確保により増大
2. 応用範囲：形式変換に伴う利用分野の制限
3. 曲面品質：圧縮後の品質確保

以上の問題点を解決するために，本研究では，曲面データをNURBS形式のままで扱い，かつ曲面の境界部での不連続性を回避する圧縮手法を構築する．

3　境界曲線網を用いたNURBS曲面圧縮

3.1　手法の概要

本研究で提案するNURBS曲面データの圧縮手法は，境界曲線を転送し，データの受け取り側で曲面を内挿する．また，内挿曲面と元曲面との間に生じる差異に応じて，差分データを用いた修正を施すことで再現性を

Compression of NURBS Surfaces by Interpolating Boundary Curves and Using Surface Difference
Yoshiyuki Furukawa, School of Engineering, The University of Tokyo
Hiroshi Masuda, Research into Artifacts, Center for Engineering, The University of Tokyo

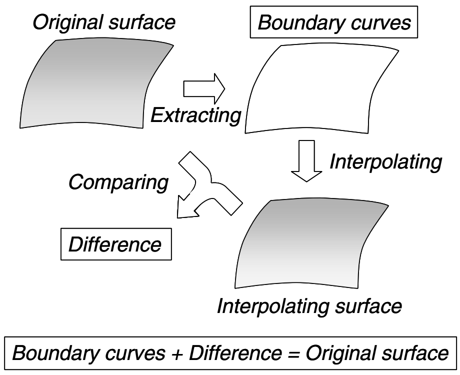


図 1. 提案する圧縮手法の基本概念

高める．提案する手法の基本的な概念を図 1 に示す．これは，ある曲面を境界曲線による内挿曲面と差分データによって表現することを意味する．

本手法の特徴の一つは，NURBS 曲面の境界曲線網を転送データとして，受け取り側で曲面を内挿することによりデータ量を減少させる点である．そのために，NURBS 曲線で定義された境界曲線網から，一意に NURBS 曲面が決定する内挿法を定義する．境界曲線を固定することによって，前節で述べた問題点の一つである境界の不連続性を回避できる．また，境界曲線データ自体に対する圧縮操作を定義し，それを用いることでデータ量を削減する．

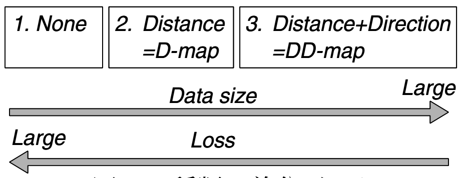


図 2. 3 種類の差分データ

もう一つの特徴は，状況に応じた再現性とデータ量の調整を可能とする差分データの導入である．差分データとは元の曲面と内挿曲面の差異を数値化し，それを元に内挿曲面を修正するためのもので，図 2 に示すように三種類定義する．一つは差分データを用いた修正を行わない場合で，これは境界曲線の内挿のみで曲面を表現することになる．残りの二つは，差異の距離成分である単一の実数集合からなるデータおよび，距離だけでなく方向成分を含めた複数の実数集合からなるデータの二つを用いる．本手法では，前者の距離成分のみで表現する差分データを D-map，後者の方向成分を含む差分データを DD-map と呼ぶことにする．各差分データは 2 次元離散コサイン変換を用いた圧縮を施して転送データとする．

3.2　全体の流れと処理構成

図 3 に提案する圧縮手法の全体の流れを示す．全体の流れは大きく分けて圧縮部と伸長部の二つから構成される．圧縮部では，元の曲面データを境界曲線と差分データに分割し，それぞれに対する圧縮を行う．伸長部では，境界曲線に関する圧縮データと差分に関す

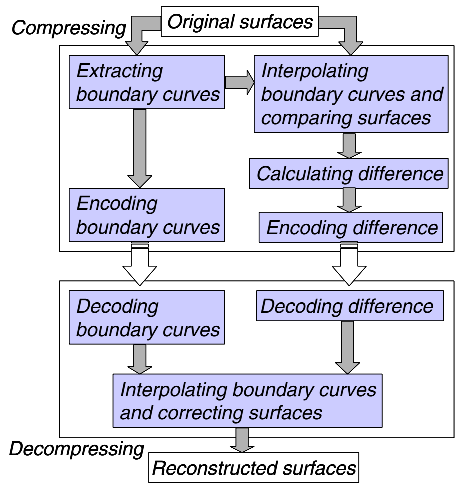


図 3. 提案手法による圧縮と伸長の流れ

る圧縮データを入力として，曲面を内挿して修正を施すことで曲面データを復元する．

4　境界曲線の圧縮と内挿曲面の生成

4.1　境界曲線網の抽出

曲面データから境界曲線を抽出するには，単に境界の制御点を抜き出す．複数の曲面で構成されるデータにおいては，曲面間の接続情報を利用して，曲線の重複を排除した境界曲線網を抽出することができる．

対象とする NURBS 曲面 $\mathbf{S}$ の定義式を示す．

$$\mathbf{S}(u,v)=\frac{\sum_{i=0}^{n}\sum_{j=0}^{m}N_{i,p}(u)N_{j,q}(v)w_{ij}\mathbf{P}_{ij}}{\sum_{i=0}^{n}\sum_{j=0}^{m}N_{i,p}(u)N_{j,q}(v)w_{ij}}\ . \tag{1}$$

ここで，$\mathbf{P}_{ij}$ は制御点座標，w_{ij} はウェイト，p,q は次数，$(n+1),(m+1)$ は制御点数，N は B スプライン基底関数である．

また，ノットベクトル $\mathbf{U},\mathbf{V}$ は次のように定義する．

$$\begin{aligned}\mathbf{U}&=\{u_i\}\ \ (i=0,\cdots,n+p+1)\\&=\{\underbrace{0,\cdots,0}_{p+1},u_{p+1},\cdots,u_n,\underbrace{1,\cdots,1}_{p+1}\}\,,\\\mathbf{V}&=\{v_j\}\ \ (i=0,\cdots,m+q+1)\\&=\{\underbrace{0,\cdots,0}_{q+1},v_{q+1},\cdots,v_m,\underbrace{1,\cdots,1}_{q+1}\}\,.\end{aligned} \tag{2}$$

一方，NURBS 曲線の定義は次式とする．

$$\mathbf{C}(u)=\frac{\sum_{i=0}^{n}N_{i,p}(u)w_i\mathbf{P}_i}{\sum_{i=0}^{n}N_{i,p}(u)w_i}\ . \tag{3}$$

曲面 $\mathbf{S}$ が与えられたとき，抽出の候補となる境界曲線は 4 本で，それぞれを $\mathbf{C}_i(i=0,\cdots,3)$ とすると，次

式で表される．

$$\begin{aligned}&\mathbf{C}_0(u)=\mathbf{S}(u,0)\,,\,\mathbf{C}_1(v)=\mathbf{S}(0,v)\,,\\&\mathbf{C}_2(u)=\mathbf{S}(u,1)\,,\,\mathbf{C}_3(v)=\mathbf{S}(1,v)\,.\end{aligned}\tag{4}$$

ただし，曲線 $\mathbf{C}_0$，$\mathbf{C}_2$ のノットベクトルは $\mathbf{U}$ であり，$\mathbf{C}_1$，$\mathbf{C}_3$ のノットベクトルは $\mathbf{V}$ である．

4.2 境界曲線の圧縮

境界曲線の転送時には，曲線の端点を固定した圧縮を行う．曲線の 2 端点間を線分で結び，各制御点をその線分の内分点からの差分表現として扱う．境界曲線の差分表現を図 4 に示す．得られた差分表現に対して 1 次元の離散コサイン変換 (DCT) を用いた圧縮を施す．

境界曲線 $\mathbf{C}$ が与えられたとき，端点 $\mathbf{P}_0$，$\mathbf{P}_n$ を結ぶ線分を等分する内分点 $\mathbf{R}_i(i=0,\cdots,n-2)$ は次式で表される．

$$\mathbf{R}_i=\left(1-\frac{i+1}{n}\right)\mathbf{P}_0+\frac{i+1}{n}\mathbf{P}_n\,.\tag{5}$$

各内分点 $\mathbf{R}_i$ と，それに対応する制御点 $\mathbf{P}_{i+1}$ 間の差分 $\mathbf{V}_i$ を次式で定義する．

$$\begin{aligned}\mathbf{V}_i&\equiv(V_i^x,V_i^y,V_i^z)\\&=\mathbf{P}_{i+1}-\mathbf{R}_i\,.\end{aligned}\tag{6}$$

同様に，ウェイトに関しても差分 V_i^w を定義する．

$$V_i^w=1-w_i\,.\tag{7}$$

全ての差分 $V_i^\alpha(i=0,\cdots,n-2;\alpha=x,y,z,w)$ に対し，1 次元の離散コサイン変換 (DCT) を施す．

$$D_k^\alpha=\sqrt{\frac{2}{n-1}}C_k\sum_{i=0}^{n-2}V_i^\alpha\cos\frac{(2i+1)k\pi}{2(n-1)}\,,\tag{8}$$

ただし，C_k は次式で決定する．

$$C_k=\begin{cases}\frac{1}{\sqrt{2}} & (k=0)\\ 1 & (k\neq 0)\,.\end{cases}\tag{9}$$

得られた DCT 係数配列 D_k^α に対して，次式による a ビット整数への量子化を施す．ただし，関数 $Round$ は

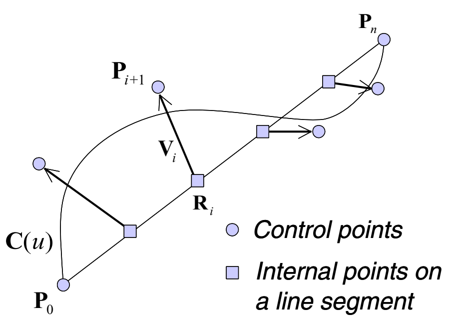


図 4. 圧縮のための境界曲線の差分表現

最も近い整数を返すとする．

$$\begin{aligned}&\bar{D}_k^\alpha=Round\left(\frac{D_k^\alpha}{Q_k}\right)\,,\\&Q_k=\frac{(1+k)D_0^\alpha}{2^a}\,.\end{aligned}\tag{10}$$

滑らかな曲線においては，量子化された DCT 係数は高周波成分になるほど 0 の比率が多くなるという性質がある．また，ウェイトが全て 1 の場合は係数は全て 0 になる．そこで，0 についてのランレングス符号化を行い，n 個の連続した 0 を 0n のように書くことでデータ量を削減する．

4.3 内挿曲面の生成

抽出された境界曲線を元に内挿曲面を生成する手法について述べる．曲面の内挿には双 3 次ブレンドクーンズ法を用いる．ここでは Lin ら [15] の定式化を用いる．この手法は境界曲線 $\mathbf{S}(u,0)$，$\mathbf{S}(u,1)$，$\mathbf{S}(0,v)$，$\mathbf{S}(1,v)$ と境界曲線上で定義される境界横断導関数 $\mathbf{S}_v(u,0)$，$\mathbf{S}_v(u,1)$，$\mathbf{S}_u(0,v)$，$\mathbf{S}_u(1,v)$，さらにツイストベクトル $\mathbf{S}_{uv}(0,0)$，$\mathbf{S}_{uv}(1,0)$，$\mathbf{S}_{uv}(0,1)$，$\mathbf{S}_{uv}(1,1)$ を元に NURBS 形式の双 3 次ブレンドクーンズ曲面を生成する．これらの内挿条件を図 5 に示す．

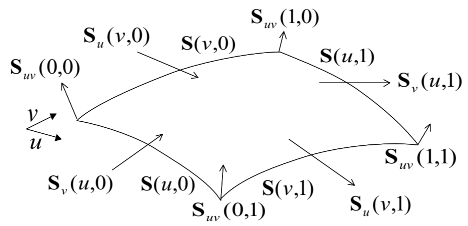


図 5. 双 3 次ブレンドクーンズ法の内挿条件

圧縮データには境界曲線のみを含め，境界横断導関数とツイストベクトルは境界曲線網から計算する．以上の条件が得られたとき，内挿曲面 $\mathbf{S}^I(u,v)$ は次式のように求められる．

$$\mathbf{S}^I(u,v)=\mathbf{S}_1^I(u,v)+\mathbf{S}_2^I(u,v)-\mathbf{S}_3^I(u,v)\,.\tag{11}$$

ここで，$\mathbf{S}_1^I$ は $\mathbf{S}(u,0),\mathbf{S}(u,1),\mathbf{S}_v(u,0),\mathbf{S}_v(u,1)$ による v 方向三次のベジエブレンド曲面で，$\mathbf{S}_2^I$ は $\mathbf{S}(0,v),\mathbf{S}(1,v),\mathbf{S}_u(0,v),\mathbf{S}_u$(1,v) による u 方向三次のベジエブレンド曲面，$\mathbf{S}_3^I$ は四隅の点と各点における u 方向，v 方向の接線ベクトル，ツイストベクトルによる双三次ベジエ曲面である．

5 差分データを用いた曲面修正

5.1 差分表現の定義

ここでは，元の曲面と内挿曲面の差異を数値化した差分データ表現の定義について述べる．

まず，曲面間の差異を詳細に数値化した表現として，DD-map の定義を述べる．図 6 にその概要を示す．DD-map は元の曲面と内挿曲面が持つ制御点とウェイトの

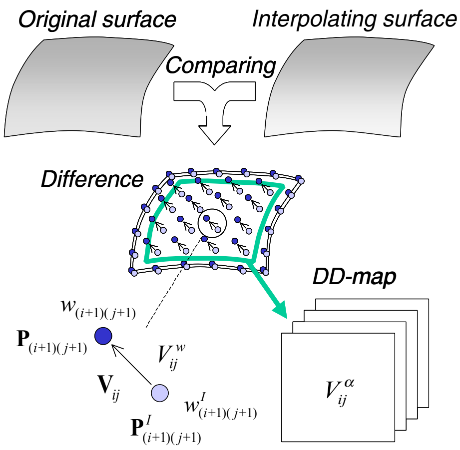


図 6. DD-map の定義

差分を直接用いる．ただし，境界では曲面が一致しているため，境界を除く内側の領域を対象とする．

元の曲面 $\mathbf{S}$ と内挿曲面 $\mathbf{S}^I$ の差分は各曲面の持つ制御点 $\mathbf{P}_{ij}, \mathbf{P}^I_{ij}$ とウェイト w_{ij}, w^I_{ij} の差分集合で表現できる．曲面の境界がすべて一致していることから，制御点間の差分集合 $\mathbf{V}_{ij}(i = 0, \cdots, n-2; j = 0, \cdots, m-2)$ を次式で定義する．

$$\begin{aligned}\mathbf{V}_{ij} &\equiv (V^x_{ij}, V^y_{ij}, V^z_{ij}) \\ &= \mathbf{P}_{(i+1)(j+1)} - \mathbf{P}^I_{(i+1)(j+1)} \,.\end{aligned} \tag{12}$$

同様に，ウェイトの差分集合 $V^w_{ij}(i = 0, \cdots, n-2; j = 0, \cdots, m-2)$ を次式で定義する．

$$V^w_{ij} = w_{(i+1)(j+1)} - w^I_{(i+1)(j+1)} \,. \tag{13}$$

これらをまとめて差分集合 $V^{\alpha}_{ij}(i = 0, \cdots, n-2; j = 0, \cdots, m-2; \alpha = x, y, z, w)$ とし，これを DD-map と定義する．DD-map は 4 種類の実数による 2 次元配列の集合である．これは差分を厳密に表現するが，データ量が多いという特徴を持つ．

次に，DD-map より少ないデータ量で差分を表現する D-map について述べる．図 7 にその概要を示す．少ないデータ量で表現するために，既知の量である内挿曲面上の法線ベクトルを方向ベクトルとして計算した移動距離を用いる．この移動距離を計算するために，内挿曲面の制御点から法線ベクトル方向に制御点を配置した近似曲面を計算する必要がある．また，DD-map と同様に境界を除く内側の領域を対象とする．

元の曲面を近似した曲面 $\mathbf{S}^A$ を次式で定義する．

$$\mathbf{S}^A(u, v) = \sum_{i=0}^{n} \sum_{j=0}^{m} N_{i,p}(u) N_{j,q}(v) \mathbf{P}^A_{ij} \,. \tag{14}$$

この近似曲面 $\mathbf{S}^A$ の制御点 $\mathbf{P}^A_{ij}$ は内挿曲面 $\mathbf{S}^I$ の制御点 $\mathbf{P}^I_{ij}$ から $\mathbf{n}_{ij}$ の方向にあると定義する．よって次式が成り立つ．

$$\mathbf{P}^A_{ij} = \mathbf{P}^I_{ij} + d_{mi+j} \mathbf{n}_{ij} \,, \tag{15}$$

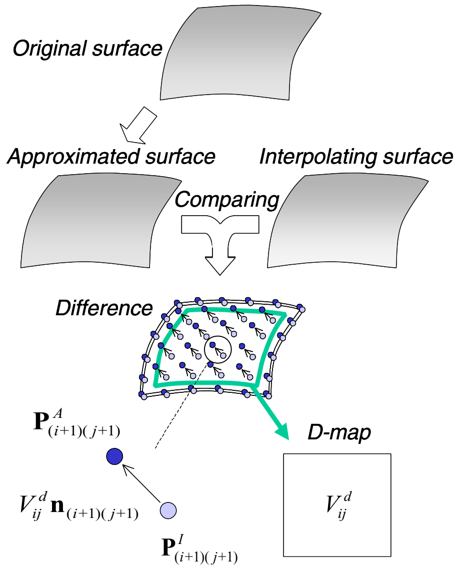


図 7. D-map の定義

ただし，d_{mi+j} は各制御点間の距離である．

DD-map の場合と同様に，曲面の境界部分は一致しているものとして，差分 $V^d_{ij}(i = 0, \cdots, n-2; j = 0, \cdots, m-2)$ を次式で定義する．

$$V^d_{ij} = d_{m(i+1)+j+1} \,. \tag{16}$$

この V^d_{ij} を D-map と定義する．

5.2 制御点の再配置

ここでは，元の曲面 $\mathbf{S}$ に対して，内挿曲面 $\mathbf{S}^I$ の制御点 $\mathbf{P}^I$ における法線方向に制御点を再配置した近似曲面 $\mathbf{S}^A$ を求めるための方法を示す．

境界曲線から得られた内挿曲面 $\mathbf{S}^I$ の制御点 $\mathbf{P}^I_{ij}$ に対して，曲面上の法線 $\mathbf{n}_{ij}$ を対応づける．この法線は次式で求められる．

$$\mathbf{n}_{ij} = \frac{\mathbf{S}^I_u(\hat{u}_i, \hat{v}_j) \times \mathbf{S}^I_v(\hat{u}_i, \hat{v}_j)}{|\mathbf{S}^I_u(\hat{u}_i, \hat{v}_j) \times \mathbf{S}^I_v(\hat{u}_i, \hat{v}_j)|} \,. \tag{17}$$

ただし，法線 $\mathbf{n}_{ij}$ と制御点 $\mathbf{P}^I$ を対応づけるためのサンプリングパラメータ $\{\hat{u}_i\}, \{\hat{v}_j\}$ は，$\mathbf{S}^I$ のノットベクトル $\mathbf{U}$，$\mathbf{V}$ から次式のように決定する．

$$\begin{aligned}\hat{u}_i &= \frac{1}{p} \sum_{k=i+1}^{p} u_k \,, \\ \hat{v}_j &= \frac{1}{q} \sum_{l=j+1}^{q} v_l \,.\end{aligned} \tag{18}$$

D-map を求めるために，評価関数 E を定義し，E の最小化問題に帰着する．

$$E = \sum_{k=0}^{n} \sum_{l=0}^{m} |\mathbf{S}^A(\hat{u}_k, \hat{v}_l) - \mathbf{S}(\hat{u}_k, \hat{v}_l)|^2 \,. \tag{19}$$

ただし，サンプリングパラメータ $\{\hat{u}_k\}, \{\hat{v}_l\}$ は式 (18) を用いる．

$\frac{\partial E}{\partial d_{mi+j}} = 0$ (i=0,$\cdots$,n;j=0,$\cdots$,m) から以下の式を得る．

$$\begin{aligned}\sum_{k=0}^{n}\sum_{l=0}^{m}\sum_{s=0}^{n}\sum_{t=0}^{m} & F_{ijstkl} d_{ms+t} \mathbf{n}_{ij}\cdot\mathbf{n}_{st} \\ &= \sum_{k=0}^{n}\sum_{l=0}^{m} N_{i,p}(\hat{u}_k)N_{j,q}(\hat{v}_l)\mathbf{G}_{kl}\cdot\mathbf{n}_{st}\ ,\end{aligned} \tag{20}$$

ただし，

$$\begin{aligned}F_{ijstkl} &= N_{i,p}(\hat{u}_k)N_{j,q}(\hat{v}_l)N_{s,p}(\hat{u}_k)N_{t,q}(\hat{v}_l)\ , \\ \mathbf{G}_{kl} &= \mathbf{S}(\hat{u}_k,\hat{v}_l) - \mathbf{S}^I(\hat{u}_k,\hat{v}_l)\ .\end{aligned} \tag{21}$$

これらすべての d_{ms+t} に関する式を連立して解く．

5.3 差分データの圧縮

D-map は単一の実数による 2 次元配列，DD-map は 4 種類の実数による 2 次元配列で表現されることから，2 次元離散コサイン変換 (DCT) を用いた圧縮法を用いる．

D-map と DD-map をあわせて $V_{ij}^{\alpha}(i = 0,\cdots,n-2, j = 0,\cdots,m-2, \alpha = x,y,z,w,d)$ と表し，用いる DCT は次式で定義する．

$$\begin{aligned}D_{kl}^{\alpha} = & \sqrt{\frac{2}{n-1}}\sqrt{\frac{2}{m-1}}C_k C_l \\ & \sum_{i=0}^{n-2}\sum_{j=0}^{m-2} V_{ij}^{\alpha}\cos\frac{(2i+1)k\pi}{2(n-1)}\cos\frac{(2j+1)l\pi}{2(m-1)}\ ,\end{aligned} \tag{22}$$

ただし，C_k, C_l は式 (9) を用いる．得られた DCT 係数行列 D_{ij}^{α} に対して，次式に示す a ビット整数への量子化を行う．ただし，関数 $Round$ は最も近い整数を返すとする．

$$\begin{aligned}\bar{D}_{kl}^{\alpha} &= Round\left(\frac{D_{kl}^{\alpha}}{Q_{kl}}\right)\ , \\ Q_{kl} &= \frac{(1+k+l)D_{00}^{\alpha}}{2^a}\ .\end{aligned} \tag{23}$$

曲線の場合と同様に高周波成分に 0 の比率が多くなるため，0 についてのランレングス符号化を行いデータ量を削減する．

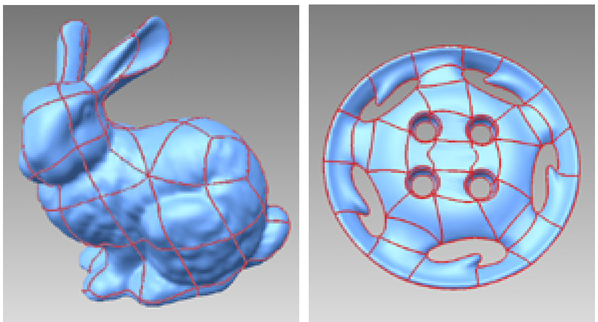
6 圧縮データ形式

圧縮データは大きく分けて，境界曲線データと曲面差分データで構成される．曲線データ自体にも端点を固定した圧縮操作を施すため，曲線データは端点の座標データと曲線差分データに分けられる．曲面間の接続情報を利用して端点や曲線データの重複を除去し，最終的な圧縮データは以下の通りである．

1. 境界曲線網の頂点座標データ列
2. 各曲線の制御点以外のデータ列
3. 各曲線の制御点に関する圧縮データ
   (a) 頂点のインデックス
   (b) 制御点以外のデータのインデックス
   (c) 1 次元 DCT 圧縮データ
4. 各曲面の差分圧縮データ
   (a) 差分データの種類
   (b) 差分データ：2 次元 DCT 圧縮データ

これらのデータをテキスト形式で出力し，汎用のテキスト圧縮を用いた符号化を行う．

7 実装と評価



(a) bunny　　(b) wheel

図 8. Example data

本研究では提案する手法を実装し，データ例を用いて検証実験を行った．図 8 に用いた 2 つの形状データの表示画面を示す．両方とも 63 枚の双 3 次 NURBS 曲面で構成されており，各曲面の制御点数はほぼ 10 × 10 である．データはテキスト形式で保存されており，位相情報は含まず曲面データのみで構成される．曲面の接続情報は処理中に内部的に作成される．

実験では複数の許容誤差を与えて，それに対する圧縮率を評価した．差分データの種類と量子化ビット値は，許容誤差の範囲内で最小のデータ量となるものを選択した．表 1 に各許容誤差に対する圧縮率の結果を示す．ただし，許容誤差はデータが持つ 3 次元座標値の誤差の絶対値とし，圧縮率は元のテキスト形式データ量に対する圧縮後のデータ量の割合とした．検証の結果，許容誤差に応じた高い圧縮率を実現していることがわかる．圧縮後のデータから曲面を再構成したときの品質を評価することで，アプリケーションに応じた許容誤差を設定して圧縮率を調整することができる．

また，処理時間に関する実験を行った．実行環境は Pentium III 1GHz，メモリ 512MB，OS RedHat7.2，コンパイラ gcc 2.96 を用いている．評価項目は各圧縮データの伸長時間と，差分データを DD-map のみに限定した場合の圧縮時間とした．これは，圧縮時に差分データの種類を選択するためには曲面間の誤差計算が必要であり，その計算コストが大きく実時間性がないことによる．一方，伸長時にはその計算が不要なため，実時間性が確保できる．よって，差分データを選択する方式でデータ圧縮を行う際には，あらかじめ圧縮データを計算して用意し，必要なときに転送して伸長することが現実的である．表 1 の結果から，伸長時には差分データ数の割合が影響を与えていることがわかる．これは例えば，D-map が内挿曲面の法線ベクトルを利用するため，その計算時間の影響等が原因である．伸長時間に比べて圧縮時間が多くなっているが，これは圧縮時には許容誤差に応じた最適な量子化ビット値の計算や，データの重複除去を行うことによる．

さらに，曲面数が増えた場合の処理時間を表 2 に示す．ここでは前述のように差分データを DD-map に限定したときの処理時間を示している．曲面数が増加したとき，伸長時に対して圧縮時にかかる時間の増加が

表 1. 実験結果

(a) bunny

許容誤差	データ量 テキスト形式	符号化後	圧縮率	差分データ数 None	D-map	DD-map	処理時間 圧縮時	伸長時
(元データ)	553.1 kB	163.8 kB	29.6 %					
1×10^{-4}	125.7 kB	52.3 kB	9.5 %	0	0	63	0.60 秒	0.14 秒
1×10^{-3}	103.0 kB	40.2 kB	7.3 %	0	0	63	0.61 秒	0.14 秒
1×10^{-2}	79.8 kB	27.7 kB	5.0 %	0	4	59	-	0.15 秒
5×10^{-2}	56.3 kB	18.5 kB	3.3 %	1	25	37	-	0.23 秒
1×10^{-1}	43.9 kB	14.7 kB	2.7 %	5	34	24	-	0.25 秒

(b) wheel

許容誤差	データ量 テキスト形式	符号化後	圧縮率	差分データ数 None	D-map	DD-map	処理時間 圧縮時	伸長時
(元データ)	598.5 kB	196.0 kB	32.8 %					
1×10^{-3}	129.0 kB	53.0 kB	8.9 %	0	0	63	0.58 秒	0.16 秒
1×10^{-2}	105.1 kB	40.0 kB	6.7 %	0	0	63	0.56 秒	0.16 秒
1×10^{-1}	71.2 kB	24.3 kB	4.0 %	0	25	38	-	0.18 秒
5×10^{-1}	44.0 kB	15.1 kB	2.5 %	20	29	24	-	0.26 秒
1×10^{-0}	39.0 kB	13.5 kB	2.3 %	33	23	7	-	0.25 秒

著しいことがわかる．これは，圧縮データ作成時に行うデータの重複を除去する処理が曲面数増加によって著しく増加することが原因である．曲面の制御点数が増えた場合も一枚あたりの処理時間は増加するが，曲面数増加による影響に比べれば問題にならないと思われる．

表 2. 曲面数と処理時間

	データ量	曲面数	圧縮時	伸長時
bunny	553 kB	63	0.6 秒	0.1 秒
A	2564 kB	318	5.0 秒	0.8 秒
B	8209 kB	1104	68.4 秒	3.5 秒

8 まとめ

本研究は 3 次元形状データの有効活用を念頭に置いた形状データ圧縮の基礎的研究として，NURBS 曲面モデルを対象とした圧縮手法を構築した．構築した手法は，NURBS 曲面の境界曲線網を転送し，受け取り側で曲面を内挿することでデータ量を削減している．また，元の曲面と内挿曲面の差異に応じて 3 種類の差分データから 1 種類を選択し転送し，かつ境界曲線と差分データに対する DCT 圧縮を用いることにより，データの利用目的に応じた圧縮率と品質の調整を可能とした．

今後は圧縮率の向上と処理時間の短縮を対象として研究を行う予定である．

参 考 文 献

[1] Deering, M.: Geometry compression, *Proceedings of ACM SIGGRAPH'95*, pp. 13-20 (1995)

[2] Taubin, G., Rossignac, J.: Geometry compression through topological surgery, *ACM Transactions on Graphics*, Vol. 17, No. 2, pp. 84-115 (1998)

[3] Gumhold, S., Straßer, W.: Real time compression of triangle mesh connectivity, *Proceedings of ACM SIGGRAPH'98*, pp. 133-140 (1998)

[4] Rossignac, J.: Edgebreaker: Connectivity compression for triangle meshes, *IEEE Transactions on Visualization and Computer Graphics*, Vol. 5, No. 1, pp. 47-61 (2000)

[5] 増田宏, 大渕竜太郎, 青野正樹: 位相操作を用いた 3 次元形状モデルのデータ圧縮法, 情報処理学会論文誌, Vol. 39, No. 7, pp. 2189-2195 (1998)

[6] Hoppe, H.: Progressive meshes, *Proceedings of ACM SIGGRAPH'96*, pp. 99-108 (1996)

[7] Li, J., Kuo, C.-C.: Progressive compression of 3D graphic models, *Proceedings of IEEE ICMCS'97*, pp. 135-142 (1997)

[8] Taubin,G., Guéziec, A., Horn, W., Lazarus, F.: Progressive forest split compression, *Proceedings of ACM SIGGRAPH'98*, pp. 123-132 (1998)

[9] Pajarola, R., Rossignac, J: Compressed progressive meshes, *IEEE Transactions on Visualization and Computer Graphics*, Vol. 6, No. 1, pp. 79-93 (2000)

[10] Khodakovsky, A., Schröder, P., Sweldens, W.: Progressive geometry compression, *Proceedings of ACM SIGGRAPH2000* , pp. 271-278 (2000)

[11] Balan, R., Taubin, G.: 3D mesh geometry filtering algorithms for progressive transmission schemes, *Computer-Aided Design*, Vol. 32, pp. 825-846 (2000)

[12] DeVore, R.A., Bjorn, J., Lucier, B.J. : Surface compression, *Computer Aided Geometric Design*, Vol. 9, pp. 219-239 (1992)

[13] 増田宏, 大渕竜太郎, 青野正樹: 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, Vol. 40, No. 3, pp. 1188-1195 (1999)

[14] 脇田玲, 矢島誠, 原田毅士, 鳥谷浩志, 千代倉弘明: ラティス構造に基づく軽量で高品質な Web3D データ表現, 情報処理学会論文誌, Vol. 42, No. 5, pp. 1170-1181 (2001)

[15] Lin, F., Hewitt, W. T.: Expressing Coons-Gordon sufaces as NURBS, *Comuputer-Aided Design*, Vol. 26, No. 2, pp. 145-155 (1994)