

## 3次元ウィジェットを用いた変形ツールの構築

栗山 繁<sup>†</sup> 橋 啓八郎<sup>†</sup>

本研究では、物体の幾何形状を変形するための対話ツールを、3次元作業空間での3次元ウィジェットの直接操作により構築する手法を提案する。本手法により、3次元格子状に配置された点群の動作がアフィン変換を用いた拘束条件で指定され、ウィジェットを用いた簡略かつ直観的な直接操作で種々の複雑な変形操作が構成される。

### Construction of Deformation Tools with 3D Widgets

SHIGERU KURIYAMA<sup>†</sup> and KEIHACHIRO TACHIBANA<sup>†</sup>

We propose a method of constructing interactive tools for deforming geometric shapes through direct manipulations of three dimensional widgets in a three dimensional working space. This method specifies the behavior of the points arranged in a three dimensional lattice by using the constraints of affine transformations, and it composes various complicated manipulations of deformation through simple and intuitive interactions with the widgets.

#### 1. はじめに

画像生成用プロセッサの処理速度の向上による3次元グラフィックスの実時間表示が実現したことにもない、画面上での対話的な直接操作によって3次元形状を設計する手法が数多く提案されてきた。特に複雑な曲面形状を設計する手法として、制御点や曲線あるいは濃度球などの幾何要素を画面上で編集して形状を設計する手法<sup>1),2)</sup>と、いったん設計した形状データに変形を施す手法<sup>3)</sup>が提案されている。既存の形状に変形を施して新たな形状を設計する手法は、その大半が変形の数学モデルの提案であり、多くの手法は変形に用いる操作の自由度が少なく、変形パターンの種類も限定されている。

一方、グラフィカル・ユーザ・インタフェースにおけるウィジェットと呼ばれる対話操作のための部品を、3次元的な形状とその振舞いに拡張した3次元ウィジェットと呼ばれる部品を用いて、直接操作によって変形を施すツールを構築する環境<sup>4),5)</sup>が開発された。しかしそれらの環境は、CADやCG等のシステム開発者のための、直接操作のツールの実装を支援するものである。ゆえに、変形ツールの導入および拡張はもっぱら開発者の手に委ねられており、プログラミングのでき

ないデザイナーは、あらかじめ用意された変形ツールのみを用いての設計作業を強いられ、自ら変形の機能を拡張することはできない。

近年、画面上での対話的な直接操作のみによって、拘束条件を有する3次元ウィジェットを構築する手法<sup>6)</sup>が提案された。しかしこの3次元ウィジェットは、3次元の形状に対してアフィン変換等の線形な変形を施すものであり、複雑な曲面の形状を設計するためには、より複雑な非線形の変形が必要とされる。

3次元空間での変形のパターンを柔軟に構成できる手法として、Free-From Deformation (以後、FFD)と呼ばれる数学モデルが提案されている<sup>7)~10)</sup>。この手法は、制御点を3次元格子状に配置して変数空間を定義し、その制御点を操作して変数空間内部に包含される幾何形状を変形させるものであり、変形を任意の解析曲面やポリゴンデータ等に、そのトポロジに関係なく適用でき、格子点の数によって変形の複雑度を増減できるという利点を有する。しかしこの手法は、格子を構成する多数の制御点を操作することによって変形の具合を調節することから、たとえばスプライン曲面を定義する2次元配置の制御点を操作するのと同程度の煩雑な作業が要求されることになる。

マウス等の2次元ポインティング・デバイスは、制御点を操作する際にその位置を平面上で移動させるのには適しているが、平面上に拘束されない点の位置を3次元空間で指定するには、3面図等の複数の2次元

<sup>†</sup> 広島市立大学情報科学部  
Faculty of Information Sciences, Hiroshima City University

座標系を用いなければならず、そのような操作に熟練するには訓練が必要とされる。ゆえに、制御点を3次元空間で直接操作する3次元ウィジェットを提供することにより、操作環境を簡略化できる。また、制御点が形状に影響を及ぼす範囲は他の制御点の影響範囲と重なっているのが一般的であり、制御点群の部分集合を一定の規則で同時に操作する方が、各点を個別に操作するよりも形状を直観的に制御しやすくなることが考えられる。

以上の考察に基づき、制御点群を3次元空間で効率良く操作するための3次元ウィジェットに対して、以下の機能を提案する。

- 3次元格子を構成する制御点の複数個を同時に操作できる。
- 同時に操作される制御点の動作に関して拘束条件を設定できる。
- 制御点の動作およびその拘束条件を容易に変更できる。

本研究の目的は、これらの機能を実現するような3次元ウィジェットを3次元空間における直接操作により構成するシステムを開発し、直観的かつ簡略な変形ツールを作成する環境を提供することである。このシステムにより、プログラミングの知識を持たないユーザにでも、種々の変形操作を可能にする独自の3次元ウィジェットが柔軟に構成できるようになる。

## 2. 複合3次元ウィジェットの構築

### 2.1 幾何胞体および対話操作のウィジェット

我々は、制御点を操作する3次元ウィジェットに必要とされる機能を実現するために、以下に列挙する手法を提案する。

- 制御点とその隣接関係を表す幾何胞体を複合的に3次元ウィジェット化し、それらとの対話操作により制御点の動作を指定する。
- 幾何胞体を表す3次元ウィジェットにアフィン変換を施すための3次元ウィジェットを導入する。また、対話時におけるウィジェット間の動作の拘束条件を直接操作により指定する。
- 3次元ウィジェットの階層的な構成を直接操作により指定する。

まず最初に、格子状に配置された制御点およびその隣接関係を定義する幾何胞体(点, 辺, 面)を3次元ウィジェット化したもの(以後, セル・ウィジェット)と、それらにアフィン変換を施す対話部品を3次元ウィジェット化したもの(以後, アフィン・ウィジェット)を導入して、制御点の動作を指定する。

点や辺および面を表すセル・ウィジェットの位置, 向き, 大きさは, それらの基準形に対するアフィン変換要素で表され, その変換要素はセルを構成する制御点の位置より計算される。アフィン・ウィジェットは, 平行移動, 回転移動, 拡大縮小のアフィン変換を施す対話オブジェクトとして設計され, セル・ウィジェットに装着してその位置や向き, および大きさを変化させる。ただし, 実際にはアフィン・ウィジェットはセル・ウィジェットを構成する制御点の位置を変化させ, その結果としてセルのアフィン変換要素の値が更新されてセルの位置や向き等が変化する。制御点の作業空間上での位置は, 初期的に設定される基準位置に対するアフィン変換行列で決定され, この変換行列はアフィン・ウィジェット自身の位置, 向き等を決定するアフィン変換要素と, アフィン・ウィジェットを直接操作して得られるアフィン変換の局所的な変動量より計算される。ゆえに, 制御点の平行移動の向きや回転移動の回転軸などはアフィン・ウィジェット自身の位置や向きで決定されることとなるが, これらの値を更新するには, アフィン・ウィジェットの位置や向きを直接操作する別の対話部品を使用する。

また, アフィン・ウィジェットは, 直接操作の対象外のときは, その位置や向き等を決定するアフィン変換要素が, 装着しているセル・ウィジェットのアフィン変換要素の値に影響され, そのセルの動作に追従するようになる。これは, アフィン・ウィジェットがセル・ウィジェットに装着されると同時に, 両ウィジェット間にアフィン変換要素の拘束条件による束縛関係が設定されるためであり(ウィジェット間の束縛に関しては, 次節で詳しく述べる), これらの拘束条件により, セル・ウィジェットに対するアフィン・ウィジェットの相対的な位置関係を, 操作を通じて一定に保つことが可能となる。

図1に, 3次元格子点とその幾何胞体を構成するセル・ウィジェットの構成例を示す。ただし, 表示の複雑さを避けるため, 面を表すセル・ウィジェットは選択された場合にのみ線画で表示される。また, 図2(p.1986 参照)に図1のセル・ウィジェットに対してアフィン・ウィジェットを装着した例を示す。この例では, 矢印と円環で表される平行移動と回転移動のアフィン・ウィジェットが, 同色のセル・ウィジェットに対して装着されている。なお図2は, 正規空間( $0 \leq x, y, z \leq 1$ )で定義されたセル・ウィジェットに対して, 赤の面のセル・ウィジェットに装着されたアフィン・ウィジェットをz軸回りに正方向から見て反時計回りに約5度回転移動させ, 緑の辺のセル・ウィ

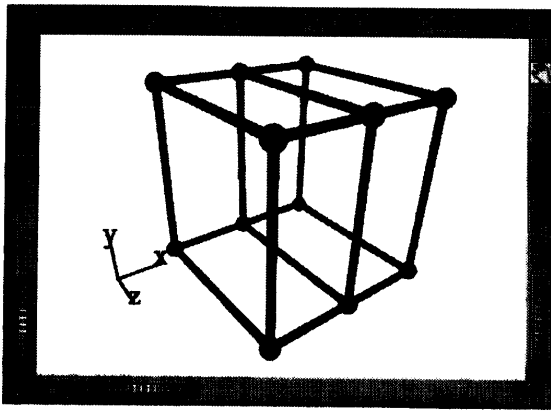


図1 格子を構成するセル・ウィジェットの生成例  
Fig.1 An example of cell widgets representing a lattice.

ジェットに装着されたアフィン・ウィジェットを  $x$  軸の正方向に約 0.5 平行移動させ、青の点のセル・ウィジェットに装着されたアフィン・ウィジェットを  $z$  軸の正方向に約 0.25 平行移動させた例である。

## 2.2 ウィジェット間の束縛

1つの操作で多数の制御点を規則的に同時に移動させる場合、制御点間に拘束条件を設定する必要がある。辺や面を表すセル・ウィジェットは、そのセルを構成する制御点間の拘束条件を設定しているが、これだけでは隣接する制御点間だけしか同時に移動できない。ゆえに、ウィジェット間に束縛関係を指定することによって、より広い範囲にわたる制御点間に拘束条件を設定する。このとき束縛関係を指定されたウィジェットの種類によって、以下の効果もたらされる。

- セル・ウィジェット対アフィン・ウィジェット  
アフィン・ウィジェットを操作することによって、セル・ウィジェットにアフィン変換が施される。アフィン・ウィジェットが操作されていない場合は、セル・ウィジェットのアフィン変換要素に従ってアフィン・ウィジェットに変換が施される。この際、アフィン変換の平行移動、回転移動、および拡大縮小の各要素に関して、セル・ウィジェットからの拘束の有無を指定できる。ただし、セルの種類によりデフォルトの値が設定され、たとえば点を表すセル・ウィジェットに対しては、平行移動の要素のみがアフィン・ウィジェットへの拘束条件として初期設定される。
- アフィン・ウィジェット対アフィン・ウィジェット  
一方のアフィン・ウィジェットを操作することにより、その変動量が他方のアフィン・ウィジェットの変動量に伝播する。アフィン変換要素の種類が異なる場合には適当な変換則が自動的に適用さ

れ、たとえば平行移動のウィジェットと回転移動のウィジェットが束縛関係にある場合は、平行移動の距離が回転移動の回転角に変換される。ユーザは変動量を変換する機構の入出力関係と、変換の際に用いられる諸定数の値を2次元ウィジェットを用いて指定し、変換則を柔軟かつ詳細に設定できる。

- セル・ウィジェット対セル・ウィジェット  
一方のセル・ウィジェットに変換が施されることにより、そのアフィン変換要素が他方のセル・ウィジェットのアフィン変換要素に伝播する。この場合もセル・ウィジェット対アフィン・ウィジェットの場合と同様に、アフィン変換の各要素に対する拘束条件を各束縛関係ごとに設定できる。

これらの効果により、1つのアフィン・ウィジェットを操作するだけで、複数のアフィンおよびセルのウィジェットが動作するので、制御点を複合的に同時に移動させることが可能になる。より具体的に述べると、形状を変形するツールを構成するうえで望ましい、以下に列挙する環境が実現できる。

- 対称性を有する変形を行うウィジェットを容易に構成できる：  
面対称の位置にあるセル・ウィジェット間に束縛関係を指定したり、それらに装着した同種のアフィン・ウィジェット間に束縛関係を指定することにより、面対称な変形が構成できる。
  - 工作機械のアフォーダンスを用いたウィジェットを構成できる：  
回転移動のアフィン・ウィジェットと平行移動のアフィン・ウィジェットに束縛関係を指定すると、ハンドルを回すことによって上下動の圧搾を行うプレス機や万力等の操作を模倣できる。
  - ユーザ・インタフェースの一貫性を保持できる：  
セル・ウィジェットとアフィン・ウィジェットの両方に対して同様の操作で束縛関係を指定できるので、両ウィジェットの違いを意識せずに一貫性のある操作で変形ツールを構成できる。
  - 柔軟かつ多様に拘束条件を設定できる：  
セル・ウィジェット間の束縛関係により、それらを構成するすべての制御点につねに同様の動作をもたらす静的な拘束条件が設定できる。一方、アフィン・ウィジェット間の束縛関係により、各ウィジェットの位置、向き等の状態によって異なる動作をもたらす動的な拘束条件が設定できる。
- ## 2.3 格子の複合ウィジェットとその階層化
- 3次元格子点の幾何胞体を構成するセル・ウィジェッ

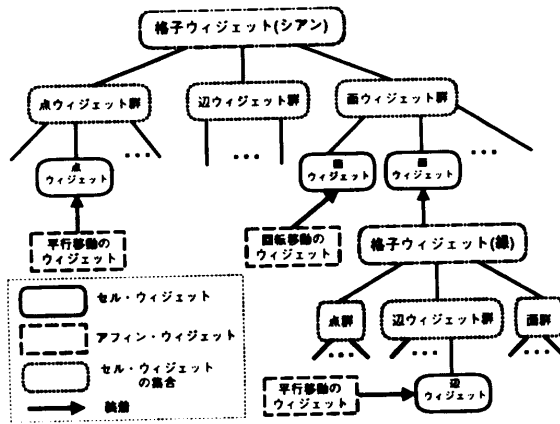


図4 図3のウィジェットに対する木構造データ  
Fig. 4 Tree structured data representation for Fig. 3.

トの集合とそれらに装着されたアフィン・ウィジェットの集合は、格子点を操作するために複合的なウィジェット（以後、格子ウィジェット）を構成する。この格子ウィジェットが変形操作の最小単位となるが、それに含まれる各ウィジェットは、他の格子のウィジェットと2.2節で述べた束縛関係を任意に指定できる。形状設計の実際の作業には複数の格子ウィジェットを用いることが考えられるが、それらの格子を構成するセルおよびアフィンのウィジェット間に束縛関係を指定することにより複合的な変形機構を実現できる。

また本システムでは、格子ウィジェットをアフィン・ウィジェットの拡張として実装し、2.2節で述べたように格子ウィジェットと他の格子ウィジェットに含まれるセル・ウィジェット間に束縛関係を設定する。この束縛関係により、前者の格子ウィジェットを後者の格子ウィジェットに含まれる対話部品として機能させ、格子ウィジェットの階層的な表現を可能とする。たとえば、キャラクタ・デザインにおける顔面形状のような設計には、曲面自身を階層的に定義して複雑な形状を設計する手法<sup>11)</sup>が効果的なので、本システムの3次元ウィジェットにも階層的な構成を導入する。

図3 (p.1986 参照) に、階層的に定義された格子ウィジェットの構成例を示す。この例では、緑の格子ウィジェットが、シアン色の格子ウィジェットに含まれる緑の面のセル・ウィジェットと束縛関係にある。またウィジェットの階層構造は、システム内では図4で示されるような、木構造のデータで保持される。

### 3. システムの実装

#### 3.1 ウィジェットの実装

セル・ウィジェットやアフィン・ウィジェットは、その位置や向き、および大きさを決定するアフィン変換

行列を内部変数として保持する。このアフィン変換行列の値は、束縛関係にある相手のウィジェットのアフィン変換行列の値と、自身のウィジェットが操作された際に更新されるアフィン変換行列の積として計算される。ゆえに、ウィジェットを操作するイベントが発生した際に、自身の変換行列の値とともに、束縛関係にあるウィジェットの値が更新される。ただし、束縛する側のウィジェットのアフィン変換行列は、各束縛関係に対して設定された拘束条件に従って平行移動、回転移動等の成分に分解され、束縛される側のウィジェットのアフィン変換行列を更新する。

アフィン・ウィジェットのアフィン変換行列の値は、自身が操作された際の局所的な座標空間におけるアフィン変換の変動量を、大域的な世界座標系での値に変換するのに用いられる。一方、格子ウィジェットもアフィン・ウィジェットと同様に構成されるが、格子ウィジェットが内部変数として保持するアフィン変換行列は、格子の局所的な座標空間を世界座標系での値に変換するのに用いられる。

アフィン・ウィジェット間に束縛関係が設定された場合は、そのアフィン変換の局所的な変動量を保持する内部変数の値が伝播して同時に更新されるようになる。ただし、アフィン・ウィジェットの種類（すなわち、アフィン変換の型）が異なる場合は、内部変数の値がデータ・タイプを変換する機構を通して（たとえば、平行移動距離の変動量が回転角の変動量に変換される）更新され、種類の異なるアフィン・ウィジェットが同時に動作する。このとき、種類の異なるアフィン・ウィジェット間の入出力関係、および変動量の変換則を柔軟に設定できる機構を有するオブジェクトが生成される。

#### 3.2 変形操作の数学モデル

FFDの数学モデルは、3次元格子状の制御点の位置座標  $P_{i,j,k}$  を3変数の基底関数の係数に用いた以下の式で表される。

$$X_{FFD} = \sum_{i,j,k} B_i(u) B_j(v) B_k(w) P_{i,j,k} \quad (1)$$

ただし、基底関数  $B_l(x)$ ,  $l = i, j, k$ ,  $x = u, v, w$  は通常、バーンスタイン (ベジエ) 多項式やB-スプラインが用いられる。

FFDを用いた変形操作は、変形される物体の形状を定義する点を3次元格子内での変数空間における局所的な座標値  $(u, v, w)$  に埋め込む計算と、その局所座標値  $(u, v, w)$  を点  $X_{FFD}$  に式 (1) を用いて写像する計算に2分される。局所的な座標値  $(u, v, w)$  への埋め込みの計算は、物体を構成する各点  $Q_n$  に対

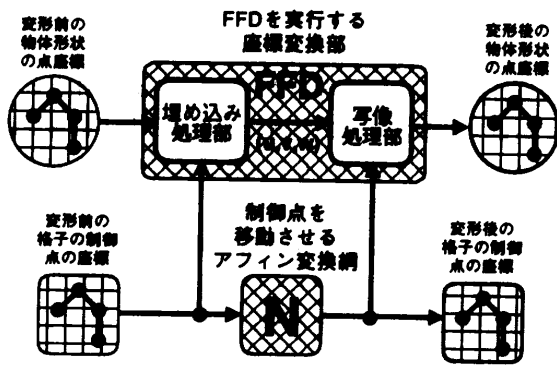


図5 FFDの処理機構  
Fig. 5 A mechanism of FFD.

して、

$$Q_n = \sum_{i,j,k} B_i(u_n) B_j(v_n) B_k(w_n) \bar{P}_{i,j,k} \quad (2)$$

を満たす座標値  $u_n, v_n, w_n$  を計算する。ただし、 $\bar{P}_{i,j,k}$  は変形前の制御点の基準位置座標とする。基底関数がバースタイン多項式で、 $\bar{P}_{i,j,k}$  が直交座標上に等間隔に配置されている場合は、式(2)は線形方程式を解いて計算されるが、他の基底関数を用いた場合や格子の点が不等間隔に配置された場合、あるいは円筒放射状に格子の点が配置された場合などは、式(2)の計算は非線形方程式を数値的かつ近似的に求めることになり計算量も増大する。しかしながら、格子点を操作して変形を行っている際には局所座標値  $u_n, v_n, w_n$  は変化しないので、式(2)の局所座標値の計算は、FFDの格子と変形する物体の相互の位置関係が決まった際に一度だけ計算すればよい。

以上の観点から、FFDの機構を局所座標への埋め込みを計算する処理部分と、写像点を計算する処理部分に分けて実装する。我々はこの埋め込みと写像の処理部分をイベント駆動型のデータ処理を行うオブジェクトとして設計し、FFD全体の処理を、これらのオブジェクトを用いて図5に示すような機構で実現する。このとき埋め込みの処理部は、変形する物体に対するFFDの制御点の初期配置が決まったときに値が更新され、写像の処理部はウィジェットによる対話操作により制御点が位置を変化させるごとに値が更新される。

一方、FFDの各制御点のウィジェットの操作ともなう位置の座標は、アフィン変換網と呼ばれるイベント駆動型の処理機構によって計算される。この処理機構は、セル・ウィジェットにアフィン・ウィジェットを装着した際とウィジェット間に束縛関係を設定した際に、アフィン変換行列の値を計算するオブジェクトを動的にネットワーク状に結合して構成される。

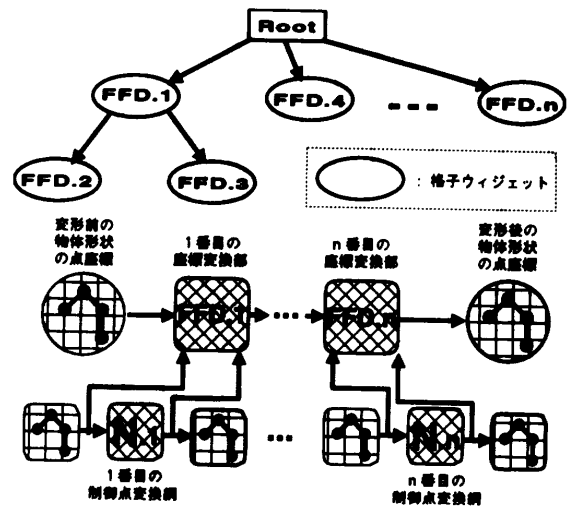


図6 階層的な格子のFFDパイプライン処理化  
Fig. 6 A pipeline process of FFDs for hierarchical lattices.

2.3節で述べたように、格子を階層的に構成した場合、変形する物体上の点が複数の格子内の変数空間に含まれるようになるので、ある格子に対する物体の変形後の出力座標値を他の格子の変形前の入力座標値として、各点の変形後の座標値を逐次計算する必要がある。本システムでは、木構造で表現される階層的な構造に対して、各格子のウィジェットを図6に示すように行きかけ順で探索しながら、各格子に対するFFDの処理機構を直列にパイプライン化する。このとき、階層関係において親に相当する格子に関する変形が先に行われ、その格子の対話部品として含まれる子の格子に関する変形が後に行われる。同レベルの階層にある格子間の優先順位は、実際にはウィジェットの設計者によって対話的に指定され、それがデータの木構造に反映される。

変形機構をパイプライン化した場合、パイプラインの後段の格子の埋め込み処理部の値はその前段の格子の写像処理部の値と同じ頻度で更新されるので、前段の格子を操作する際の計算量が増大してしまい、対話操作の実時間応答性に支障が生じる。ゆえに、全体の計算量を軽減するために、操作の対象となっている格子以外のFFDの埋め込み処理部の値の更新の頻度を調節できるようにする。

### 3.3 システム開発環境

我々は、以上に述べてきたシステムを開発するためのツールとして、Open GL上に開発されたOpen Inventor<sup>12),13)</sup>を用いた。Open Inventorは、オブ

☆ Open GLおよびOpen InventorはSilicon Graphics, Inc.の登録商標です。

ジェクト指向のグラフィクス・アプリケーション構築ツールであり、3次元物体の形状および属性を定義するシーン・オブジェクトと呼ばれるクラス・ライブラリに加えて、3次元物体のアフィン変換を行う対話型のオブジェクトである *SoManipulator* や、その構成部品である *SoDragger*、およびイベント駆動型のデータ処理を行うアニメーション・オブジェクトである *SoEngine* 等を提供する。我々のウィジェット構築システムでは、この *SoDragger* や *SoEngine* に対する C++ のクラス・ライブラリを拡張して、制御点の操作を行う3次元ウィジェットおよび FFD の処理機構のクラス・ライブラリを実装した。また、アフィン・ウィジェットを直接操作するツールとして、*SoManipulator* で構成される対話部品を用いた。

Open Inventor は、幾何形状や面の属性および光源などのデータと同様に、対話モデルに関しても標準的なデータ・ファイル形式を定義しているの、変形ツールのデータが他の3次元グラフィクス・データと明確な区別なくデータベース化できるという利点を有する。また、Open Inventor のデータ・ファイル形式は計算機の機種に依存しないように設計されているので、この形式を用いることにより変形ツールの異機種間での可搬性、互換性が達成される。

また、本研究のシステムは Silicon Graphics 社のワークステーション Indigo-2 を用いて開発された。

#### 4. 変形ツールの作成

本システムを用いて3次元ウィジェットを構成し、幾何物体に変形を施す一連の操作を以下に詳説する。

- (1) 3次元格子を構成するセル・ウィジェットの生成：  
ダイアログ・ボックスにより格子の各軸に沿う制御点の数を指定すると、その格子点および幾何胞体を構成するセル・ウィジェット群が ( $0 \leq x, y, z \leq 1$ ) の正規空間上に生成される。
- (2) アフィン・ウィジェットのセル・ウィジェットへの装着：  
セル・ウィジェットを画面上でピックアップした後、メニューでアフィン・ウィジェットの種類を選択すると装着が実行される。
- (3) アフィン・ウィジェットと格子ウィジェットの状態の指定：  
アフィン・ウィジェットまたは格子ウィジェットを画面上でピックアップすると、それに対してアフィン変換を施す対話部品が自動的に装着され、その部品を画面上で直接操作してウィジェットの

位置や向き、および大きさを調節する。また、画面上の何もない場所をピックアップすると、装着された対話部品が自動的に脱着される。

- (4) ウィジェット間の束縛関係の指定：  
画面上で2つのウィジェットをピックアップして束縛メニューを実行すると、ピックアップされたウィジェットの種類で決まる束縛関係が指定される。
- (5) 拘束条件を決めるアフィン変換要素の設定：  
束縛されている側のウィジェットを画面上でピックアップすると、その地点で設定されている拘束条件をトグル・ボタンで示すメニューが表示されるので、それを用いてアフィン変換の各要素ごとに拘束の有無を再設定できる。
- (6) ウィジェットの幾何物体への装着および脱着：  
幾何物体と格子ウィジェットを画面上でピックアップした状態で、装着メニューを実行すると、その格子ウィジェットの座標空間が幾何物体の外接直方体と一致するように変換され、幾何物体を構成する点の座標が FFD の処理機構に取り込まれ、アフィン・ウィジェットの直接操作により形状を変形できる状態になる。また、格子ウィジェットを画面上でピックアップした状態で脱着メニューを実行すると幾何物体の点が FFD の処理機構から切り離される。
- (7) ウィジェットのファイル入出力：  
幾何形状や属性のデータと同様に、ダイアログ・ボックスでファイル名（変形ツールの名前）を入力して、ファイル入出力を実行する。

以上の操作は、3次元作業空間における3次元ウィジェットのピックアップおよび直接操作を基本に行われるが、数値およびテキスト入力が必要される操作や特殊な命令の実行に対しては、ダイアログ・ボックスやアプリケーション・ボタン、およびメニュー・パネル等の2次元ウィジェット (Motif<sup>14</sup>) により実装) を使用する。

本システムは、変形ツールの作成過程と変形操作の実行過程における状態の遷移を意識させない、モードレスな対話環境を提供しているの、変形動作を指定してウィジェットを設計する作業と、ウィジェットを操作して装着された幾何物体を変形する作業が並行して行える。すなわち、上記の(2)~(6)の操作は順序に関しての依存性がなく独立に実行できる。また、階層的な格子ウィジェットを構成する際には、複数のウィンドウ上で個別に格子ウィジェットを構成し、ある格子ウィジェットのセル・ウィジェットと他の格子ウィジェットの全体を選択した後、束縛関係を指定して構成される。

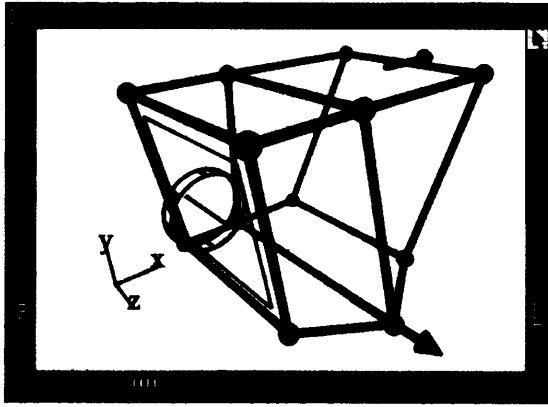


図2 アフィン・ウィジェットの装着例  
Fig. 2 An example of affine widgets connected to cell widgets.

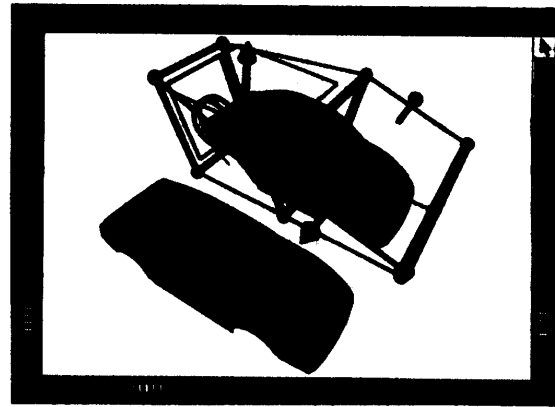


図7 3次元ウィジェットを用いた変形操作例  
Fig. 7 An example of deformation with 3D widgets.

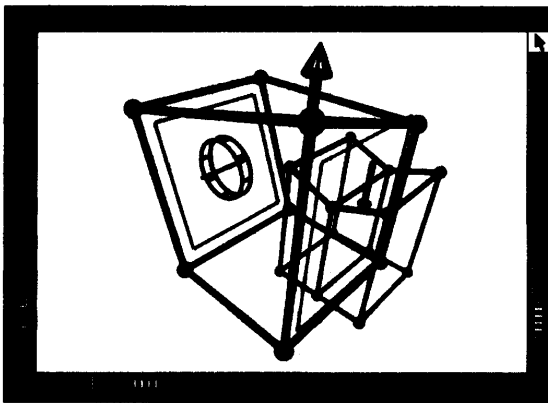


図3 階層的な格子ウィジェットの構成例  
Fig. 3 An example of hierarchical representation of lattice widgets.

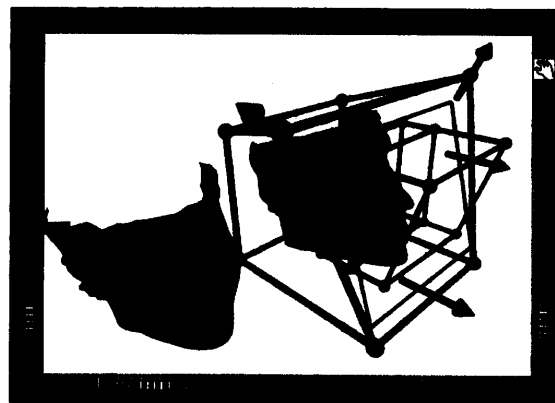


図8 階層的なウィジェットを用いた変形操作例  
Fig. 8 An example of deformation by using a hierarchical representation of lattices.

図7と図8に、3次元ウィジェットを用いて具体的な幾何形状に変形操作を施した例を示す。

図7は、 $2 \times 3 \times 2$ の制御点を有する格子を構成するセル・ウィジェットを生成し、2つの面のウィジェットに平行および回転移動のウィジェットを装着して移動の向き、回転軸を調節した後、2つの辺に対して平行移動のウィジェットを逆向きに装着してそれらに束縛関係を指定して、面対称な変形を行った例である。また図8は、階層的なウィジェットの構成による変形ツールの操作例を示しており、 $2 \times 2 \times 2$ の制御点を有する格子を構成するセル・ウィジェットの面のウィジェットに、階層的に格子ウィジェットを束縛させた例である。また、2つの点のウィジェットに装着した平行移動のウィジェットどうしと、異なる格子ウィジェットに含まれる2つの辺のウィジェットに装着した平行移動のウィジェットどうしには、各々束縛関係が指定されている。

## 5. おわりに

本論文では、3次元形状に対して複雑な変形を施すツールとしての3次元ウィジェットを、対話的な直接操作によって構成する手法について述べた。本システムにより作成される複合的な3次元ウィジェットは、アフィン変換要素を用いた拘束条件を指定することにより複数の格子点の動きを複合的に同時に制御することができる。また、視覚的かつ直接的な画面上でのウィジェットの設計を支援することにより、プログラミングの知識のないユーザにでも自分自身の変形ツールが容易に作成できる。また、ウィジェットがデータとしてファイルに保存されることから、変形ツールの再利用および共有が可能となる。さらに、本システムで構成される変形ツールは、FFDの手法のすべての利点を継承するので、時間因子を操作する3次元ウィジェットを導入することにより、アニメーションの動作の設計に対しても有効に用いられる<sup>15)</sup>。

本報告では、変形の数学モデルとしてFFDを取り上げたが、本手法は本質的にはその数学的モデルに限定されるものではなく、多数の制御点を操作対象とするすべての変形手法に適用可能である。それらの変形手法の数学モデルをコールバック関数として実装すれば、ウィジェットの作成環境をそのまま用いることができる。また、本システムで作成される3次元ウィジェットは、2次元状に配置された制御点を操作するのにも用いることが可能なので、たとえば、スプライン曲面等の形状を直接操作するツールとしても役立つ。さらに、本システムは3次元入力デバイスを利用することにより、操作性をより一層向上させる可能性があるが、これらのシステムへの実装は今後の課題である。

**謝辞** 本論文は、広島市立大学第3号特定研究である、“次世代・インダストリアルデザインシステムの開発”の研究成果をまとめたものである。特定研究を協同して行った同大学の芸術学部・デザイン工芸学科の中島健明助教授ならびに吉田幸宏講師に感謝の意を表します。

### 参 考 文 献

- 1) Farin, G.: Curves and Surfaces for Computer Aided Geometric Design, *A Practical Guide*, Academic Press (1990).
- 2) Blinn, J.F.: A Generalization of Algebraic Surface Drawing, *ACM Trans. Graphics*, Vol.1, No.3, pp.235-256 (1982).
- 3) Barr, A.H.: Global and Local Deformations of Solid Primitives, *Computer Graphics*, Vol.18, pp.21-30 (1984).
- 4) Snibbe, S.S., Herdon, K.P., et al.: Using Deformation to Explore 3D Widget Design, *Computer Graphics*, Vol.26, No.2, pp.351-352 (1992).
- 5) Strauss, P.S. and Carey, R.: An Object-oriented 3D Graphics Toolkit, *Computer Graphics*, Vol.26, No.2, pp.341-349 (1992).
- 6) Zeleznik, R.C., Herndon, K.P., et al.: An Interactive 3D Toolkit Constructing 3D Widgets, *Computer Graphics*, Vol.27, No.4, pp.81-84 (1993).
- 7) Sederberg, T.W. and Parry, S.R.: Free-form Deformation of Solid Geometric Models, *Computer Graphics*, Vol.20, No.4, pp.151-160 (1986).
- 8) Coquillart, S.: Extended Free-form Deformation: A Sculpting Tool for 3D Geometric Modeling, *Computer Graphics*, Vol.24, No.4, pp.187-196 (1990).
- 9) Lamousin, H.J. and Waggenspack, W.N., Jr.:

NURBS-based Free-form Deformations, *IEEE Computer Graphics and Applications*, Vol.14, No.6, pp.59-65 (1994).

- 10) Hsu, W.M., Hughes, J.F. and Kaufman, H.: Direct Manipulation of Free-form Deformations, *Computer Graphics*, Vol.26, No.2, pp.177-184 (1992).
- 11) Forsey, D.R. and Bartels, R.H.: Hierarchical B-spline Refinement, *Computer Graphics*, Vol.22, No.4, pp.205-212 (1988).
- 12) Wernecke, J. and Open Inventor Architecture Group: *The Inventor Mentor: Programming Object-oriented 3D Graphics with Open Inventor™*, Release 2, Addison-Wesley (1994).
- 13) Wernecke, J. and Open Inventor Architecture Group: *The Inventor Toolmaker: Extending Open Inventor™*, Release 2, Addison-Wesley (1994).
- 14) Open Software Foundation: OSF/Motif Reference Guide.
- 15) Coquillart, S. and Jancene, P.: Animated Free-form Deformation: An Interactive Animation Technique, *Computer Graphics*, Vol.25, No.4, pp.23-26 (1991).
- 16) Kuriyama, S. and Tachibana, K.: Interactive 3D Widgets for Free-form Deformation, *Computer Graphics and Applications*, pp.320-329 (1995).
- 17) 栗山 繁, 橋啓八郎: 形状デザインにおける変形ツールの構築環境, *グラフィクスとCAD研究*, 77-6, pp.35-41 (1995).

(平成8年2月23日受付)

(平成8年9月12日採録)

栗山 繁 (正会員)



昭和37年生。昭和62年大阪大学大学院工学研究科電子工学専攻修士課程修了。翌年日本アイ・ビー・エム(株)入社、東京基礎研究所にてCAGDシステムの研究開発に従事。平成6年退社。現在、広島市立大学情報科学部情報機械システム工学科助教授。計算機支援幾何設計、グラフィクス・インタフェース、対話システムに関する研究に従事。電子情報通信学会、ACM各会員。



**橘 啓八郎**

昭和13年生。昭和42年大阪大学大学院工学研究科電子工学専攻博士課程単位取得。昭和42年同大学工学部助手，助教授を経て昭和50年長崎総合科学大学工学部教授，電算機センター所長。昭和60年摂南大学工学部経営工学科教授。現在，広島市立大学情報科学部情報機械システム工学科教授。電子回路に関する電子工学，計算機の応用を中心とする情報工学の研究に従事。工学博士。電子情報通信学会，経営工学会，オフィス・オートメーション学会各会員。

---