

# 面補間機構を有する三次元形状表示装置

井越 昌紀      小西 和正  
 (財)機械振興協会 技術研究所

## 1. はじめに

機械、建築、土木をはじめとし、三次元形状を扱う分野への計算機利用の努力はめざましいものがある。三次元形状を外部表現するには、計算機の中には形状のモデルの存在が必要である。このモデルは単に表示のためのものであることも多い。この場合には時間が多少かかっても最終的に与えられる絵が出ればよく、絵自体がグラフィック製品となる。このときのモデルは絵を得ることがすべてであり、計算機内のモデルを工学的な利用に役立てることはできない。このため、工学的に三次元形状を扱う分野、すなわち、設計・生産にモデルを用いようとする分野ではモデリングの研究が追々盛んに行われた。この場合、内部モデルは幾何モデルと呼ばれ、外部表現は内部モデルを人に知らせる一つの手段となる。

幾何モデルの内部表現の方法も各種存在するようになったが、内部表現の構造により外部表現する過程も異なるのが現状である。また、内部モデルを作るとはじめて外部表現ができることは、逆に見れば内部モデルができるれば外部表現もできることになり、人の創造性を多様な面から要求し、意図する三次元形状を感覚的に捉える設計の初期段階には、このような過程を繰り返すことは時間的・計算機の使用効率から見て効果的とはいえない。

本研究は、設計・生産のために必要な三次元形状の外部表現を、設計・生産のための厳密な内部モデルを創成するフェイズと機能的・時間的に切り離し、三次元の表示モデルを独立して持たせることにより、設計・生産の計算機援助方式において、人間-機械系の対話性の向上を図ろうとするものである。本報告では、三次元形状表示を内部モデリングと切り離すための表示側に課せられる条件として、隠れ線・面消去処理、線図形・面図形の同時表示、全体形状に対する部分形状の動的表示、形状の表示集合演算の4つをあげ、それらを実現するための平面補間、形状表示演算、アライオリテ表示コントロールについて述べ、これらの機能を持つ三次元形状表示ターミナルの構成について報告する。

## 2. 表示座標系ごとのモデル操作の持つ限界

工学的に利用される三次元形状の表示は、設計・生産のための厳密な内部モデルの形状寸法のみに着目しても不十分であり、モデルの持つ拘束条件を含めたトポロジカルな関係の解釈が必要である。

さらに、表示モデルの扱う形状寸法の精度(9ビット~16ビット)と、内部モデルのそれ(24ビット~64ビット)とは異なることより、外部表現に用いられる表現空間の精度で、内部モデルの形状を正確に表現することは不可能である。しかし、ある製品形状の設計を考えたとき、内部モデルにとって重要な形状の持つ寸法の属性は、これのみを表現空間の精

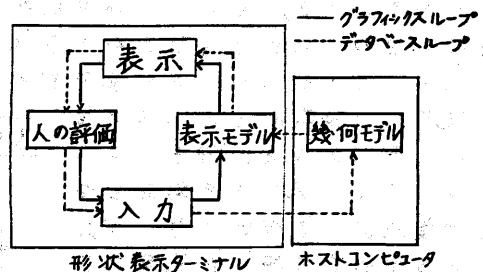


図1 グラフィックスループ

度で取り出して外部表現しても実用的に有効に用いられる。本研究における表示装置に要求する各種機能は、人の持つ低精度ではあるが全体把握に対する感覚と、機械の持つ高速な処理速度を利用し、高精度で比較的処理速度の遅い設計・生産のための内部モデル創成とのインタフェースとして働く。このためには、低精度で処理可能な入力から出力までのグラフィックスループを、データベーススループから分離し、形状操作に対する独立性を持たせる必要がある(図1参照)。このときの精度を落した寸法属性から成る形状モデルを表示モデルと呼ぶことになる。

### 3. 三次元表示装置の持つべき条件

三次元表示装置が、内部モデルを扱う装置と独立して機能するためには、いくつかの条件を必要とする。

#### 3.1 隠れ線・面消去処理

三次元形状を直観的に立体把握させるためには、一定の立体表示のための手法が必要である。この手法としては一般に透視変換、奥行き輝度変調、動的な回転表示、中間調面図形、立体視などがあるが、なかでも形状が複雑になり、曲面をも含むものになると、隠れ線・面消去処理は重要な条件である。この処理機能は一般に、内部モデルとその処理法に依存する。隠れ線・面消去処理は、多くの場合多面体を対象にして行うが、現実には形状が多面体でない場合も多い。形状が凸体である場合には比較的処理が容易であるため、非凸体(ガウス曲率が負である場所が存在する)を凸体化して処理する場合もある。しかし、曲面は凸体に分割できない。

本方式においては、形状の持つ奥行きを、ロウセルごとに比較して、目に近い方を表示させる形状の画素とするいわゆるZバッファ方式を採用してモデル依存性を極力なくした。しかし、形状によっては表示モデルの持つ精度上、どちらが目に近いか判断できない誤差範囲があり、このときには、より精度の良い内部モデルを考慮にする。

#### 3.2 線図形と面図形表示(図6参照)

形状設計において、線図形は特徴的、計量的、分析的であるから他と区別する能力に優れ、面図形は実在的、写實的、総合的であるから形状実在部の直観的伝達に優れる。一般に、線図形は抽象化された図形入出力方式で、面図形はより具体的な存在である。形状設計は立体的な環境の中での部分の付加、除去、修正が主体であることを考えると、具体的なレベルでの記述の中で、抽象化レベルでの線図形を重ね合わせて設計を行う手法が考えられる。抽象化レベルでの設計が終了したら、具体的なレベルでの面図形に変換して全体のシーンの中に埋め込む。外部表現をこの抽象化レベルを尺度にとり系列化すると図2のようになる。ここで実在的面図形とは、形状が実在的で立体的に見える最小限の輝度処理を施した面図形を言うものとする。本研究での表現の対象はモデルレベルである。

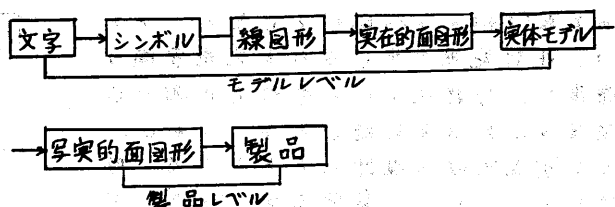


図2 外部表現のレベル

### 3.3 部分形状の動的表示

人の創造性が要求される設計の段階では、仕様に対して行ったアクションに対し、レスポンスが適度な時間内に返ってくる必要がある。抽象化レベルが進むほどレスポンスは遅くとも許される場合があるが、具体的なレベルになるに従って早い応答が望まれる。形状設計においては、部分の変更が主体であることを考えると、変更する部分のみ動的な変化が可能であれば多くの場合充分である。この部分形状の表示は抽象化レベルの高い線図形であったり、シンボルであることが多いこと、面図形であることも全体形状に対して情報量がきわめて少ないことを考え合わせると、全体と部分に分けて表示することにより、部分の動的表示が可能となる。

### 3.4 形状表示演算

形状の集合演算は、複合形状を創成するときの基本創成方式の一つであり、手続き構造である。これは、任意の形状はある基本形状を付加(和)したり、除去(差)したりしてごころとするもので、ほとんどのモデリングシステムはこの方式の入力とその処理方式を備えている。表示演算とは、形状の集合演算を表示させる部分にのみ着目して行うもので、この演算によって得られる結果は、目に最も近いところの形状が表現されているにすぎない。しかし、表示の持つ最大の目的は、人の目に触れて内部モデルを推察できる充分な情報を与えることであるから、このように表示だけでも集合演算ができることは、内部モデルの変更を併行して、手続きという人にとって分かりやすい入力形式そのものがモデルとなり、これからできるであろう内部モデルの事前の評価ができる。このことは、グラフィックスルーファが独立性を持つことができる条件の一つとなる。既報<sup>2)</sup>においては、二形状間の表示演算について述べたがあるが、本報告では手続き表現と一対一に対応して表示が得られる一般化した表示演算について考察した。

## 4. 形状表示の処理

上記の機能を満足させるため、すべての処理を画集ごとに行っている。このため本表示装置はラスタ型である。基本処理はピクセルごとであるが、これを一点ごととソフトウェアで計算しているのは多大な時間を要し、本研究の目的に合わない。ピクセルごとに処理する方式の長所は、ハードウェア処理がし易いことである。そこでハードウェア化できるための各種アルゴリズムを導出した。形状の表示モデル表現は有向境界法表現になっている。これは、表裏の区別された面の上に乗った有向線によって境界線を記述し、さらに、面を有向境界面として記述するものである。これから得られるワイヤフレーム状の中間表示モデルに対し平面補向を施しながら表示モデル全体を表現する。

## 5. 三次元平面補向

システムは表示モデルから曲面なら微小三角形群を、平面上の多角形なら有向線のつながりを平面補向機構<sup>4)</sup>と送る。ここでは、三角形の場合はそのまゝ補向を、多角形の場合は、ハードウェア凸平面補向器<sup>5)</sup>で処理できるように、任意多角形を分割しながら送る。直線補向を食め、平面補向の中の水平補向はすべてDDA方式による。

## 5. 1 主峰くずしによる任意多角形の分割

穴などがある任意多角形の三次元面補向は、多角形のぬりつがし問題の変形として考えられる。多角形のぬりつがしは、スキャンコンバーションの方法によって可能であるが、始点・終点の決定にソフトウェアが絡んでおり、速度を上げることが難しい。多角形をその節点から引いた水平線と、他の輪郭との交点を求め、得られた三角形あるいは四角形に対して水平補向を行う方法もあるが、交点を求めるのに時間がかかる。

本補向方式は凸平面補向を行うハードウェアを用意し、そのハードウェアに合うように任意多角形を効率的に分割しながら処理する。この分割法を主峰くずしによる多角形の分割と名づける。

### アルゴリズム

任意多角形は図4のようにベクトルの方向の左側に奥面があるように記述されているとする。

(1) 順位バケット創成 多角形の頂点をベクトルの方向に順に見ていき、端点を右のように4つに分類する。分類には各点が水平方向に引き延ばされたものも含まれる。このときの対象点は左側に統一する。分類の対象になった点はYの順に並べ、同じYのものはXの順に並べる。補向開始ポイントを最上位に置く。

(2) 補向開始点の決定 順位バケットを指す補向開始ポイントの位置を補向開始点とする。

(3) チェック点の決定 順位バケットを指すチェック点ポイント(初期値は最上位のつぎ)から点を取り出しチェック点とする。

(4) 補向開始 補向開始に連続されている多角形の二本の直線の左側を始線、右側を終線と見てYの値がチェック点の値になるまで、つぎつぎに輪郭線を取り出しながら凸平面補向を行う。

(5) 分岐 Yの値がチェック点と一致したら凸平面補向から出る。チェック点が現在の始線と終線の間にはさまれているものについてのみチェック点の種類によってつぎのよう処理をし、補向開始ポイントを1つ進め(2)に定める。はさまれている場合は、チェック点ポイントを1つ進め(2)に定める

(5-1) 下端点 凸平面処理の終了。最後の点または線を出す

(5-2) 分割点 始線・終線を2組に分け、新たな形状とする。つぎに左側の組みについて凸平面補向を継続する。順位バケットのなかのこの分割点の種類分けを再評価する

(5-3) 結合点 処理を終了し、新たな形状とする。順位バケットのこの結合

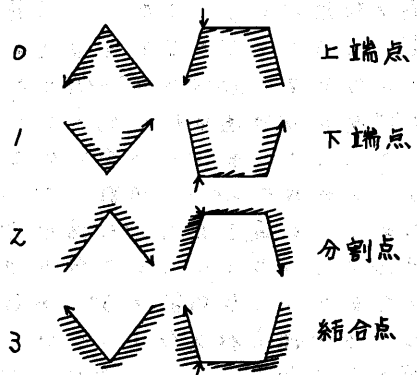


図3 端点の分類

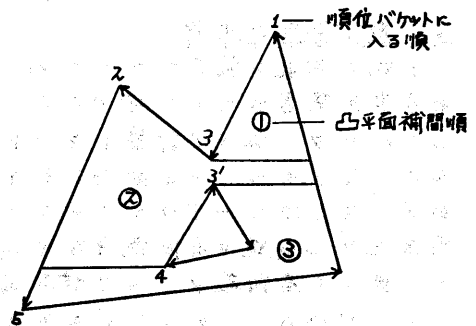


図4 主峰くずしによる多角形分割

点の種類分けを再評価する。さらに、補向途中のもう一方の端点の種類分けをし、順位バケットに入れる。

### 5.2 凸平面補間と縁取り

ここで言う凸とは、始線と終線の間に干渉点がなく、始線上の開始点から、終線上の終了点まで単純に水平補向ができる多角形を言う。この凸多角形の平面補間は、始線と終線をY方向に三次元直線補向し、Yが1つずつ進むごとにその向を三次元水平補向する。始線と終線の輝度を中間のそれよりも高くすれば縁取り線となる。このとき、輪郭線は直線補向として出すから、平面補向により得られる深さ(Z)と輝度(B)の値と異なる。このため、輪郭線を補向する際に直線補向で得られるZとBの値とは別に、平面式の増分を直線補向のX、Yの増加と同時にインクリメントに計算する処理を行わせる。このことにより、平面上のZとBの値の精度を高く保つことができる。

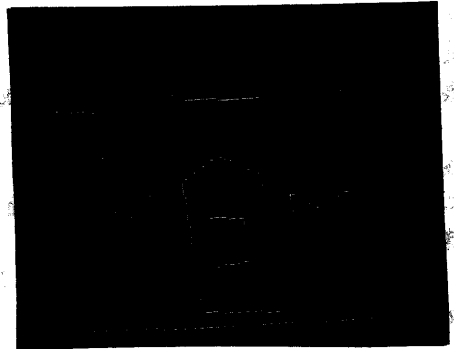


図5 多角形の平面補間



図6 線図形と面図形の干渉

### 5.3 隠れ線・面消去処理との関係

平面補向は単に面内部のみりつぶし処理だけでなく、深さ方向の補向も同時に行っている。深さをZバッファに蓄えこおくことにより隠れ面処理は単純に行なえる。このとき、縁取り線のみの高輝度にし、内部を輝度ゼロにすれば、隠れ線消去処理となる。

## 6. 形状表示演算

形状の集合演算は形状の創成の主要な方法の一つである。モデリングシステムは、この手続きをモデルとしてそのまま持っているものと、拘束条件を評価してトポロジカルなネットワーク表現として持っているものがある。内部モデルの表現形態からすれば、評価されたネットワークモデルが最終的なものとなるが、実用上は、このネットワークモデルを常に作るのではなく、曲面などの場合処理に時間がかかるように、工学的な応用にとっても必要でない場合もある。そこで形状の集合演算を表示される部分に限って行うことにより、人の入力手段である手続き的な表現がそのまま表示モデルとして利用できることとなる。しかし、この方法はあくまでも拘束条件の評価なく行われるものであるから、設計・生産の内部モデルとして厳密な解を必要とする場合は適宜データベースにより支援する必要がある。この表示演算を二形状間に限って行うことについては既に報告した。今回はこの表示演算を一般化したアルゴリズムについて述べる。

### 6.1 形状の集合演算とその対応

形状の集合演算は、形状をSとし、Uを和集合、∩を積集合とするとき

$$S = U \cup S_i; \quad \cap = U \cap S_j; i$$

で表わされる。これは、集合演算によって創成可能な形状 \$S\$ は、任意形状の積集合で得られる \$i\$ 個の積形状の和集合として表現できることを表わしている。形状 \$S\$ は閉な曲面で構成されているものとし、視線の方向に対し、可視側と不可視側に分け、それぞれ \$S\_+\$, \$S\_-\$ と表わすものとする。さらに、\$S\_j\$ が積演算対象の形状を表わすものとし、\$S\_{jk+}\$ を \$S\_j\$ を構成するセグメント面とすると、

$$S_j = \sum S_{jk+}, \quad \bar{S}_j = S_{j-} = \sum S_{jk-}$$

などの関係がある。

ここで、\$S\_{jk+}\$ は \$S\_{jk-}\$ の向きを反した曲面の排続によって閉境界面を創成することを意味する。\$S\_{jk+}\$ は可視面、\$S\_{jk-}\$ は不可視面である。差演算は、

$$S_j - S_{j+1} = S_j \cap \bar{S}_{j+1}$$

で表わされる。

### 6.2 視線上での積集合

いま積演算 \$\cap S\_j\$ において、一本の視線 \$l\$ を考える。積演算結果は、すべての \$S\_j\$ (\$j=1...m\$) の実部の交差部分であり視線 \$l\$ においてはそのなかの最も目近（可視側）画素となる。ここで形状の実部とは、視線のながで一回の可視面と不可視面にはさまれた部分である(図7)

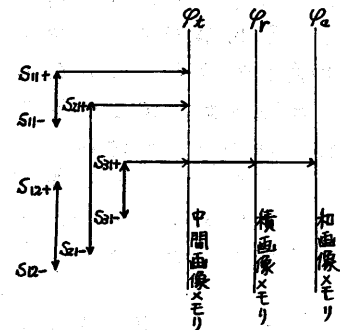
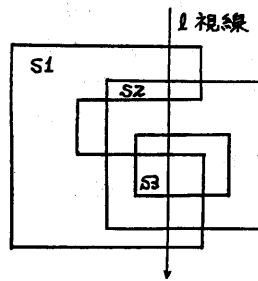


図7表示演算説明図

可視点はつきのような手順で

る。ここで、3つのバッファを考える。\$u\$ とつは最大値を入れる中間バッファ (\$Q\_b\$), \$v\$ とつは積を入れる積バッファ (\$Q\_r\$), \$w\$ とつは和を入れる和バッファ (\$Q\_a\$) である。

\$S\_j\$ のセグメント面と視線との交点のながで、積バッファ(初期値として最大値 1 を入れておく)より小さく中間バッファ(初期値として 0 を入れておく)より大きな点があれば、中間バッファの内容を入れ換える。この内容は、可視フラグ、深さおよび輝度情報を持たせるものとする。可視フラグは、そのとき入るセグメント面が可視であれば 1、不可視であれば 0 とする。このようにして、\$S\_j\$ のすべてのセグメント面 \$S\_{jk+}\$ について処理したあとで、中間バッファに蓄えられている画素が可視であれば、中間バッファから可視フラグの内容を除いた情報を積バッファに転送し、転送した結果を示す転送フラグをオンにする。

つぎに \$j\$ を 1 つ進め、同様にして積バッファより小さな深さの中で最大のものを探す。このようにして \$j\$ の範囲をループし、一つの転送フラグが立ってあとの \$n-1\$ 回すべて不可視フラグを持つ面が最大値となっていれば、処理を終了し、そのときの積バッファに蓄えられた情報がその視線での可視点となる。\$n\$ 回連続するながで、\$u\$ とつで積バッファより小さな深さを持つ面がない形状 \$S\_j\$ が存在すれば、その視線での可視点は存在しなく空集合の部分となる。

このようにして得られた可視点の情報を和バッファに和集合演算しながら送れば、視線上での一点の表示演算がごまあがる。

### 6.3 平面補面を利用した表示演算

上記の演算は一般性があり、形状の点座標を精度良く算出すれば、金型加工用

のNCラフ作成等工序上での応用も可能である。しかし、視線上での処理は速度がきわめて遅く、対話性の向上は期待できない。この視線上での処理を高速で処理可能な平面補向を利用したものに拡張した。このときのアルゴリズムは視線上のそれと同様であるが、処理の速度を上げるためにセグメント面に不活性フラグを設け、平面補向の際に、一つのピクセルも処理の対象とならなかつたら不活性フラグを立て、後続の表示演算の対象とはしない。

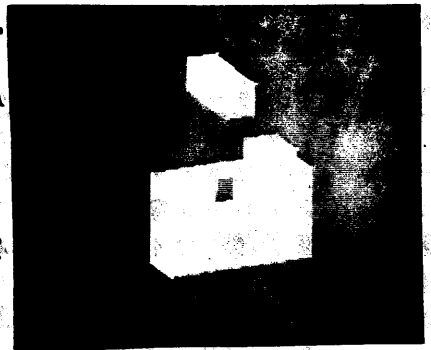


図8 表示演算例

## 7. プライオリティによる表示コントロール

平面補向により完全した深さを全画面情報に、必要に応じて形状表示演算が施され、輝度情報はフレームバッファ、深度情報はプライオリティバッファに送られる。プライオリティバッファは、累積画像に対応した深さが入っており、プライオリティの比較により、フレームバッファの書き込みをコントロールしたり、自分自身の書き込みをコントロールする。

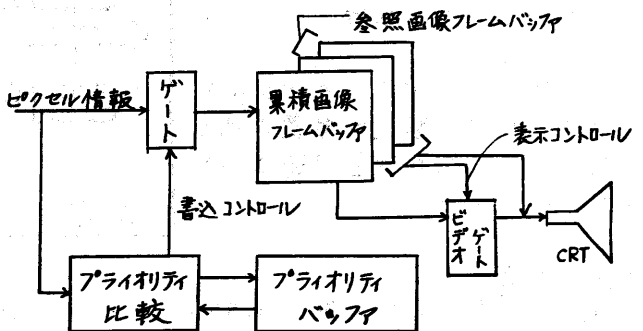


図9 プライオリティによる表示コントロール

表示される画像を入れるフレーム・バッファは、累積画像を入れるものと、参照画像を入れる二種類を用意する。ここで累積画像は全体の形状を入れるものであり、参照画像は、全体画像に働さかけようとする部分形状が線図形、面図形が入るものとする。動的な表示を得るためには、参照画像用フレームバッファを複数枚用意するとよい。

### 7.1 表示書き込みコントロール

累積画像が入った状態のところへ、新たに参照画像を送り込めると、その情報はプライオリティバッファの深さと比較され、目に近い方の参照画像のピクセルの輝度情報のみが表示書き込みコントロール信号により書き込み許可を受け参照フレームバッファに送られる。累積画像より目から遠いと書き込み禁止となる。この表示書き込みは補向と同期し、1メモリサイクル内に行われる。累積画像として付加する場合には、その参照画像をもう一度表示モデルから平面補向と表示演算を通し、プライオリティの比較を同様に行ったあと、適合したピクセルの深さ情報はプライオリティバッファの更新と、輝度情報は累積フレームバッファの更新に用いられる。

### 7.2 ビデオ信号のプライオリティ・コントロール

参照画像フレームバッファに画像があるということは、累積画像よりも優先して表示しなければならぬことを意味する。そこで、参照画像フレームバッファからのビデオ表示の際に、ピクセルごと参照画像によって累積画像の表示を禁止す

る機構を設ける。

### 8. 形状表示ターミナルとしての実装

図10に今までの述べた各機能をマイクロコンピュータ制御のインタリジエントターミナルとしてまとめ、RS232Cを通してホストコンピュータであるECLIPSE 130に接続したブロック図を示す。平面補間および表示演算はマイクロレーテンサ制御で行っており、サイクルタイムは250msである。マトリクスマルチプライアは座標変換やクリッピングに用いるもので、16ビット×16ビットの乗算チップを用い4×4のマトリクス乗算を32μsで行っている。

ホストコンピュータには厳密な内部モデルとその処理アルゴリズムを持たせる。

入力装置としては、タブレットのほか位置姿勢入力装置を持ち、直接的に形状の位置姿勢をコントロールできる。

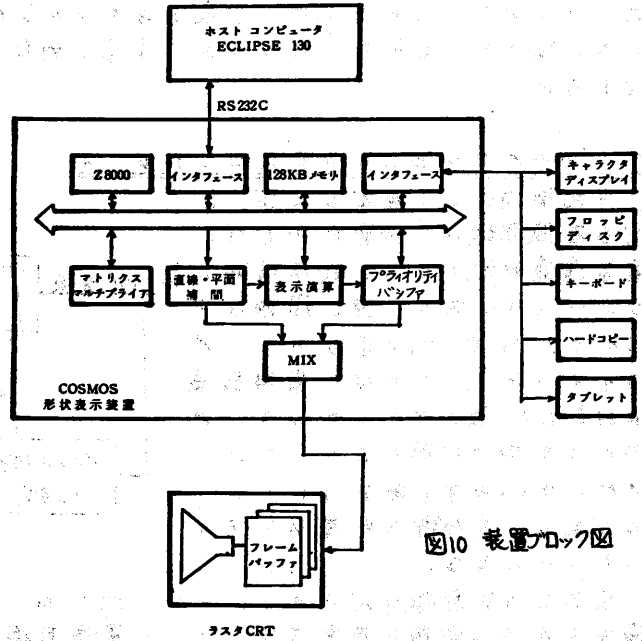


図10 装置ブロック図

### 9. おわりに

設計・生産のための形状モデルは、設計の種類とサイズによっても必ずしも厳密な内部モデルを作らなくてよい場合がある。これは加工や組立のようにシーケンスを重んじる場合や、人の感覚が配置

や美的評価が必要な形状設計の場合である。このような場合のモデルとしては、手続きとネットワークを併せ持つモデルが良いと思われる。手続きを含むモデルは外部表現に時間がかかるが、本機能によりそれがかなり高速になった。今報告では述べられなかったが、厳密なプライオリティを必要とするときの方法などについてもいくらか実験を行っており、別の機会に報告したい。

本報告は1978年より開始したCOSMOSシステム開発の成果の一部であり、ECLIPSEのマイクロプログラミングを行った小沢正照氏に深く感謝する。

### 参考文献

- 1) N. Okino, Y. Kikazu & H. Kubo : TIPS-1 Technical Information Processing System for Computer-aided Design, Proc. 2nd PROLAMAT Conf. (1973), 141
- 2) I.C. Braid : Designing with Volumes, Cantab Press, Cambridge (1973)
- 3) M. Hosaka & F. Kimura : An Interactive Geometrical Design System with Hand Writing Input, Information Processing 77, IFIP, North Holland (1977) 167.
- 4) 井越昌紀 : 面図形援助による複合形状設計(第1報) 精密機械 Vol.47, No.2 1981, p.33