グラフィクスとＣＡＤ　83－2
（１９９６．１２．１９）

ネットワークモデルを用いた３次元形状モデラ

藤村真生　小堀研一　久津輪敏郎

大阪工業大学

本論文では，空間分割モデルをＣＡＤに応用する場合に必要となる形状の幾何演算を高速に処理するための新しいデータ構造を提案する．空間分割モデルにおける形状表面のボクセルのみを相互接続することにより，データ数を大幅に削減し，かつ幾何演算を高速に処理することが可能なモデルを構築することができる．これをネットワークモデルと呼び，実際にネットワークモデルを用いて開発したモデラについて説明する．ネットワークモデルを評価するため，従来からある空間分割モデルとしてオクトツリーモデルを取り上げ，データ数，立体集合演算および幾何演算に要する処理時間の比較実験を行い評価を行った．

Geometetric modeling using NETWORK model

Masao FUJIMURA, Ken-ichi KOBORI, Toshiro KUTSUWA

Osaka Institute of Technology

We propose a new spatial partitioning representation model called NETWORK model that can realize fast geometric operation in CAD systems. It comes to possible to cut down drastic number of voxels and to reduce calculation cost, by means of defining only the voxels on the surface of a shape and their mutual connections. We also construct a three dimensional geometric modeler with the NETWORK model. To evaluate the NETWORK model and it's modeler, we made several experiments to compare the OCTREE model with the NETWORK model on the cost for geometric operations and boolean-set operations.

1. まえがき

従来よりＣＡＤの分野において物体形状を表現する際には，形状をその構成要素である点・線・面などを用いて表現する形状分割モデル[1]が多く用いられている．形状分割モデルは少ない情報により形状を表現でき，幾何演算処理に適した形状モデルである．しかし，形状演算の際に複雑な幾何要素の交差判定等の演算を行う必要があり，処理負荷・安定性に問題がある．また，ＣＡＤにおいてデザイナーが形状変形などの操作をする際にはその構成要素を意識して操作する必要があり，モデルのデータ構造に対する理解と慣れが必要である．

一方，形状を微小な立方体により表現する空間分割モデルは，位相構造を保持しない単純な形状モデルである．ところが高いモデリング精度を要求するとデータ量が莫大になり，形状操作には不向きであると考えられているため，ＣＡＤの分野においてはあまり用いられていない．しかし，データ構造が簡単で幾何要素の交差判定等の複雑な処理が不必要となるので，形状分割モデルと比較して形状操作を安定して行うことのできる形状モデルと考えられる．また，空間分割モデルをＣＡＤシステム等に応用した場合，デザイナーがデータ構造を考慮して形状操作を行う必要がなく，形状操作が容易な形状モデラとして構築できる可能性を持っている．

本稿では空間分割モデルをＣＡＤの形状モデルとして利用するために，物体形状を効率よく表現し，かつ物体形状の集合演算処理及び幾何演算処理を高速に処理するための新しい形状モデルについて考察する．提案する新しい空間分割モデルをネットワークモデルと呼び，このモデルを用いて実際に形状モデラを開発したので報告する．

2. 空間分割モデル

2.1. ボクセルモデル

ボクセルモデルは空間分割モデルの中で最も基本的なモデルである．このモデルでは形状が存在する立方体の空間を想定し，この空間をｘ，ｙ，ｚの３軸方向に均一に分割する．分割によって区切られた微小な立方体をボクセルと呼ぶ．各々のボクセルにそのボクセルが物体の内部であるか外部であるかをデータとして保持させ，形状は内部を表すボクセルの集合として表現される．

2.2. オクトツリーモデル

ボクセルモデルのデータ数を抑えるために考案されたオクトツリーモデルは，空間を再帰的に８分割してボクセルを生成する．このモデルでは形状表現に必要な部分についてのみ細かいボクセルが生成されるので，データ数を削減することが可能となる．

2.3. ネットワークモデル

本論文で提案するのは，形状表面に存在するボクセルのみを形状の情報として採用するモデル[2]である．この方法によりデータ数を最小限に抑えることが可能となるので，形状演算の際の計算回数や描画時間等を削減することができる．

空間分割モデルでは，形状の情報を離散的なデータの集合として保持する．しかし，表面のボクセルのみを保持する場合にはボクセルモデルのように離散的な状態で保持しただけでは形状に対する任意の点の内外判定ができず，立体集合演算に応用することは不可能である．この問題を解決するため，ボクセルチェーン[3]を用いる手法が提案されている．しかし，この手法は形状に穴が発生するため正確さに欠ける．また，データの検索方向が１方向であることから，ボクセルの３次元的な接続状態を判定するには多くの処理を必要とするので形状操作には不向きであると考えられる．

一方，２次元空間において効率よくデータを保持するモデルとしてはランレングス法が存在し，２次元形状のアフィン変換の手法[4]も考案されている．これを３次元形状モデルに拡張した３次元ランレングス法[5]ではランと呼ばれるｚ方向の形状データをｘ，ｙの２軸方向で保持する必要があるため，１軸方向にランを検索して変換する文献（４）の手法は適用できず，また形状データに方

向性が生じることから形状演算には不向きなものと考えられる．

ネットワークモデルでは図１に示すように形状表面のみに存在するボクセルの６近傍を双方向のポインタによって接続し，形状表面においてネットワークを構成することにより，上記の問題を解決する．ネットワークモデルは形状内部の情報を持たないので扱うボクセルの数が少なく，かつ総てのボクセルが均一のサイズであるから，形状に回転・移動を施す幾何演算に適した形状モデルとなる．また双方向のポインタを用いることから，あらゆる方向からデータを検索して容易に形状演算を行うことが可能となる．

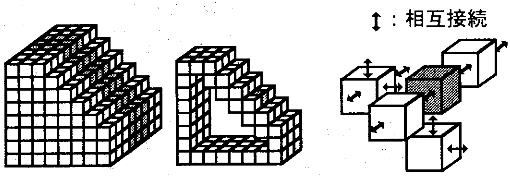


図１　ネットワークモデル

3. データ数とモデル精度

ネットワークモデルは形状表面のボクセルのみを保持するモデルである．従って，形状の表面積をS，面積あたりの精度をd，空間を各軸方向にm＝2^n分割したモデルを考えると，データ数N_Nについて次式が成立する．

$$N_N(n) = S \cdot d \propto S \cdot m = S \cdot 4^n \qquad (1)$$

従ってモデル精度を２倍にすれば，必要となるデータ数は４倍になる．一方，ボクセルモデルにおいてモデル精度を２倍にすると，必要となるデータ数は８倍，同様にオクトツリーモデルでは４倍になる．従って，ボクセルモデルと比較してネットワークモデルはオクトツリーモデルと同様にデータ効率のよい形状モデルとなる．

一般に空間分割モデルを家電製品などのデザインの初期段階で利用する場合，そのモデリングには空間の１辺の長さ約50cmに対してボクセルの１辺の長さ1mm程度の精度が必要とされている．空間分割モデルでは空間を各軸方向に２の累乗数によって分割するのが一般的であるので，この精度を実現するためには空間を各軸方向に512分割したボクセルを用いる必要がある．

4. 立体集合演算

本稿で取り上げる演算は論理積・論理和及び差演算の３種類の演算である．

ネットワークモデルは形状内部の情報を保持しないので，従来の空間分割モデルにおける立体集合演算と同様な手法によって演算を行うと形状が切断され，正常な結果を得ることはできない．そこで，ネットワークモデルの利点を損なわずに高速な立体集合演算を実現する手順を次に示す．

処理１：２つの形状の交差部分を算出する．

処理２：非交差部分のボクセルデータ群を，元の形状のうちどちらのボクセルであるか，また交差部分から見て内側，外側のどちらに存在するかという２つの状態によって４種類に分類する．

処理３：４種類に分類されたボクセルデータ群を表１に示したテーブルを参照しながら消去または交差部分と接合する．

表１　立体集合演算に用いる表

含まれる形状	内/外	積演算	和演算	差演算
形状１	内側	接合	消去	消去
形状１	外側	消去	接合	接合
形状２	内側	接合	消去	接合
形状２	外側	消去	接合	消去

処理１，２は２つの形状データを１回ずつ走査しながら並列に処理できる．また処理３は交差部分を１回走査すると処理を終了する．従って，これらの演算に要する時間T_{N1}のオーダーは，元の形状１，２のデータ数をそれぞれN_{N1}，N_{N2}とし，交差部分のデータ数をN_{NX}すると

$$\begin{aligned} O(T_{N1}) &= O(N_{N1}+N_{N2}) + O(N_{NX}) \\ &= O(N_{N1}+N_{N2}) \qquad (2) \end{aligned}$$

となる．すなわち，ネットワークモデルの集合演算の処理時間は２つの形状のデータ数の和に比例することがわかる．

5. 幾何演算

一般に３次元座標P(X,Y,Z)はアフイン変換によ

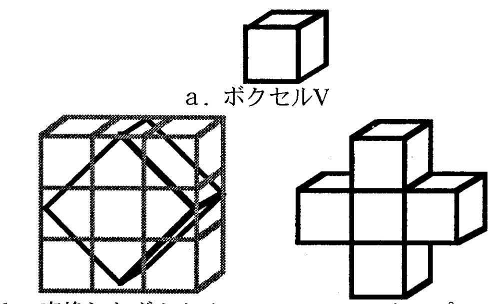
り４×４のマトリクスFを用いて次式

$$P' = FP \qquad (3)$$

によりP'(X',Y',Z')に変換される．

ネットワークモデルは形状データの位置と接続状態を保持するモデルであるので，従来の空間分割モデルと同様な方法で幾何演算を処理することは困難である．次にネットワークモデルにおいて幾何演算を実現する方法について説明する．２次元の図を用い，形状を２倍に拡大し，時計回りに４５度回転する場合のマトリクスをＦとして例にあげる．

前処理：図２ａに示したネットワークを構成するボクセル１つの形状Vに，Ｆによる幾何演算を施し，同図ｂに示した形状V'を得る．V'を図中に網掛けで示したボクセルと同じ大きさを持つ編み目によってスキャンし，その中心点で内外判定した結果は同図ｃのようになる．このボクセル群をスタンプと呼び，保持しておく．



a．ボクセルV

b．変換したボクセルV'　　c．スタンプ

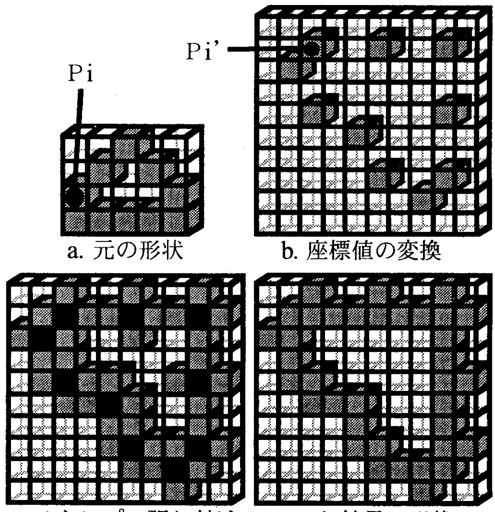
図２　スタンプの生成

処理１：例として図３ａに示したネットワーク形状を走査し，その座標値P_iをマトリクスＦを用いて変換する．ネットワークを構成するすべてのボクセルに対して新しい座標P_i' を算出すると得られる形状は同図ｂに示されるようになる．

この形状には穴が発生しており，このままでは形状を正確に表現することはできない．これらの穴は，幾何演算によって形状のサンプリング間隔とボクセルの大きさに差異を生じた場合に発生する．従って，これを回避するためには先に説明したV'のような形状のボクセルを用いればよい．しかし空間分割モデルにおいて，V'のようなボクセルを扱うことはできない．そこで本研究では，前述したスタンプを用いて穴の発生を回避する．

処理２：このスタンプを同図ｂのＰｉ' （図の網掛けのボクセル）を中心とした座標に各々配置すると，同図ｃに示したように穴が除去された形状を生成することができる．

処理３：以上の手順によって生成された形状にはスタンプの厚みが存在する．ネットワークモデルは形状表面のデータのみを保持する形状モデルであるから，結果の形状を走査して形状表面に存在するボクセルのみを得ると同図ｄに示すような目的のネットワークモデルとなる．



a. 元の形状　　b. 座標値の変換

c. スタンプの張り付け　　d. 結果の形状

図３　幾何演算の手順

処理１，２は，変換元になる形状のネットワークを走査しながら並列に処理することができる．また，処理３は結果の形状を１度走査することによって処理を終了する．従って，これらの処理に要する時間T_{N2}のオーダーは，変換元の形状に含まれるデータ数をN_{NS}，結果の形状のデータ数をN_{NR}とすれば

$$O(T_{N2}) = O(N_{NS}) + O(N_{NR}) \qquad (4)$$

となり，変換元と結果の形状の持つデータ数の和にほぼ比例した時間を要することになる．

６．形状モデラ

6.1. モデラの概要

提案したネットワークモデルを用いて，実際に形状モデラを開発した[6]．モデラはシリコン・グラフィックス社のグラフィックス・ワークステーション上で実行される．

３章で述べた精度を満たすため，本研究で開発したモデラでは１辺を５１２分割したネットワークモデルを用いた．

6.2. モデリングの機能

モデラの持つ各機能について説明する．

A．２オペランド方式による演算

モデラ内部には２個のネットワークモデルの形状を保存することが可能であり，これらによって形状演算を行う．

B．形状の読み込み・保存

作成したネットワークモデルの形状はファイルへの保存・読み込みが可能である．

C．基本形状の生成

基本形状として，ネットワークモデルで表現された直方体・球・円柱・円錐・４面体・３角柱の６種類を予め用意した．

D．形状の配置

モデラに読み込まれたネットワークモデルの形状に対して５章で述べた幾何演算を施し，所望する位置・大きさになるように配置する．

E．ネットワークモデル同士の立体集合演算

配置した２つのネットワークモデルの形状から，和・差・積の３種類の立体集合演算のいずれかにより新しい形状を作成する．

F．境界表現モデルとの立体集合演算

従来のＣＡＤシステムにより作成した形状を利用できるように，Ｂ－Ｒｅｐｓにより表現された形状をモデラ内に読み込む機能を付加した．形状は任意の位置に配置し，ネットワークモデルに対して立体集合演算を施すことが可能である．

G．ヒストリー機能

作業履歴をテキストファイルとして保存することが可能である．保存されたヒストリーファイルは編集して再び読み込み，作業を再現することができる．

7. 実験結果および考察

ネットワークモデルの有効性を検証するため，オクトツリーモデルとの比較実験を行った．

7.1. データ数

コンピュータ内に実際の形状を構築し，形状を表現するために必要となるデータ数を計測して比較する実験を行った．使用した形状は工業製品の形状を含めた６種類の形状である．結果を図４に示す．図４から，オクトツリーモデルを用いて形状を表現する場合と比較してネットワークモデルを用いた場合は３５％程度のデータ数によって形状を表現することができることがわかる．

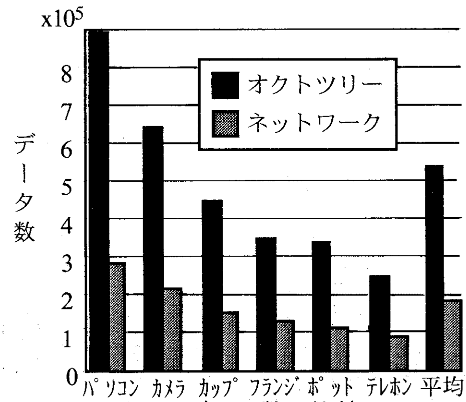


図４　データ数の比較

7.2. 立体集合演算

立体集合演算の処理時間について，以下の実験を行った．

実験１：空間を８分割した立方体と，この立方体と同じ長さの直径を持つ球形状との和演算．

実験２：実験１に用いた球形状２つの積演算．

実験３：前節のカメラ形状から実験１で用いた立方体を用いてレンズ部分を削除する演算．

実験４：図４に示した総ての形状の組み合わせによる和演算の処理時間の平均を算出した．

実験結果を表２ａに示す．なお実験に使用した計算機はIndy(150MIPS)である．この結果から，立体集合演算の処理時間はネットワークモデルにおいては最大２秒程度であり，またオクトツリーモ

デルにおいては最大３秒程度になる．従って立体集合演算においてネットワークモデルの処理時間は，オクトツリーモデルと比較して７０％程度に短縮することが可能である．

7.3. 幾何演算

前節と同様に幾何演算の処理時間について以下の実験を行った．

実験５：空間を各軸方向に４分割した立方体形状をｘ，ｙ，ｚ軸を中心にそれぞれ１５度，３０度，４５度回転する演算．

実験６：実験１で用いた球形状をｘ，ｙ，ｚの方向に球の直径と同じ距離だけ移動する演算．

実験結果を表２ｂに示す．この結果からオクトツリーモデルを用いた場合では実用的な時間で幾何演算を処理することは困難であるが，ネットワークモデルを用いた場合は数秒程度で完了することがわかった．ネットワークモデルがオクトツリーモデルと比較して幾何演算に有利な形状モデルであることが確認できる．

表２　各演算の処理時間の比較

a.立体集合演算

	O	N	N/O
実験１	0.9	0.6	67%
実験２	1.1	0.5	45%
実験３	1.3	1.1	85%
実験４	3.2	2.2	69%

b.幾何演算

	O	N	N/O
実験５	48	1.9	4.0%
実験６	392	3.8	1.0%

単位：秒，O：オクトツリーモデル
N：ネットワークモデル

7.4. モデリング

今回開発したモデラを用い，実際に形状を作成する実験を行った．作成した形状を写真１に示す．この形状のモデリングに要した時間は15分程度，演算等のモデリングの工程数は35であった．モデリングに用いた基本形状は12個である．この形状を構成するネットワークモデルのノード数は242,771である．

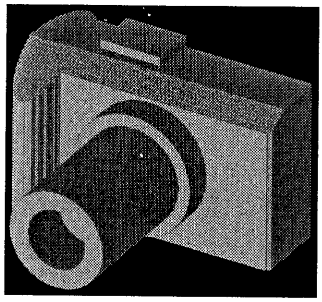


写真１　モデリング例

8. むすび

幾何演算に適した空間分割モデルとして形状表面のボクセルのみを相互接続するネットワークモデルを提案した．また，実際に対話的なモデラを開発し，モデリングの実験を行った．

ネットワークモデルは，オクトツリーモデルと比較してデータ数は３５％程度，また立体集合演算の処理時間は６８％程度に削減できることが実験により明らかとなった．幾何演算についても４％以下の時間で処理を終了することが可能となり，大幅な高速化を実現した．

開発したモデラは，空間分割モデルを用いることにより従来のＣＡＤのように形状の幾何要素を意識しながらモデリングを行う必要がなく，ユーザーインタフェイスの良いものとなった．また形状分割モデルを読み込む機能により，ハイブリッドなモデラとして構築することができた．

文　献

（１）F.Foley , A.VanDam , S.Feiner & J.Hughes : "Computer Graphic Principles and Practice Second Edition", Addison Wesley Publishing Company,pp.533-562(1990).

（２）藤村，小堀，久津輪：" 空間分割モデルにおける幾何演算についての一考察"，情処学第５１回全大（2），pp.2-267(1995)

（３）石井，小野，大和，牧野：" ボクセルチェーンを用いた3Dグラフィックス"，情処学グラフィックスとCAD研報，No.48，40-1 (1989).

（４）東海林：" ｐｘｙ表にラン形式で格納された２値画像のアフィン変換アルゴリズム"，信学論（DⅡ），J77－DⅡ，9，pp.1753-1760(1994).

（５）荒川：" 仮想空間における立体形状モデリング"，情処学グラフィックスとＣＡＤシンポジウム論文集，Vol.91，No.7，pp.33-43(1991).

（６）藤村，西尾，小堀，久津輪：" ネットワークモデルによる形状モデリング"，情処学第５３回全大（4），pp.4-51(1996)