

境界表現のための位相モデルの分類

山 口 泰†

3次元形状の境界表現は、位相情報と幾何情報を組み合わせて形状を表現する手法であり、ワイヤフレームモデルに始まり、サーフェスモデル、多様体立体モデルから非多様体位相モデルへと拡張されてきた。しかし、近年研究されてきた非多様体位相モデルという言葉は多様体立体以外の位相という意味しか持たず、表現対象の位相を構成的に定義したものではない。そのため、これまでに提案されてきた非多様体位相モデルは表現対象自体が多様であり、結果としてデータ構造なども異なっていた。そこで本研究では、各位相要素の近傍に着目することにより境界表現の位相モデルを分類し、それらの間の相違を明確にする。さらに各位相モデルのデータ構造、操作法などについて述べる。

A Taxonomy of Topology Models for Boundary Representations

YASUSHI YAMAGUCHI†

Boundary representations consisting of both topological information and geometric information can represent several classes of shapes in three dimensional space, i.e., wireframe models, surface models, manifold solid models, and non-manifold topology models. However the term, non-manifold topology, only indicates that the boundary is not 2-manifold but no more at all. Many different kinds of non-manifold topology models have been proposed. This paper is aimed at classifying the boundary representations based on the neighborhood of each point set of the topology models. Based on this classification of topology models, data structures as well as different sets of basic operations are discussed.

1. はじめに

3次元形状の境界表現は、位相情報と幾何情報を組み合わせて形状を表現する手法である。境界表現に基づく形状モデルは、ワイヤフレームモデルに始まり、サーフェスモデル、立体モデル、非多様体モデルへと拡張されてきた。従来の立体モデルは、その位相モデルとして多様体立体を扱うものであった^{1),2)}。多様体立体とは、その表面上のすべての点において、表面が2次元円盤に位相同型となるものである。すなわち、すべての稜線は2枚の面に隣接するなどの制約を満たす立体である。これは実在する立体形状を表現するには十分な表現力を持ち、かつ winged edge データ構造などの簡潔なデータ表現が利用でき、オイラー操作が定義できるなどの特徴を備えている。しかし一方で、正規化集合演算に関して閉じていないなどの問題があり、非多様体モデルの研究もさかんになされてきた^{3)~6)}。ところが、この非多様体という用語は、「多様体立体ではない」という意味しか持たず、明確に定

義された特定の位相モデルを表すものではない。したがって、これまで提案されてきた非多様体位相モデルはその表現対象自体が異なっており、結果として提案されたデータ構造なども異なっていた。

一般に非多様体モデルの特徴としてよくあげられるのは、以下の点である。

- (1) 1稜線に隣接する面の数が2つに限らない。
- (2) 1頂点に隣接する空間の数が2つに限らない。
- (3) 複数の閉じた3次元空間を表現できる。
- (4) 孤立した面や稜線をも表現できる。

立体表現としての非「2-多様体」という観点から考えると、上記のうち(1)ないし(2)が基本的な要件となる。r-set 立体^{5),7)}の扱いにおいては、まさにこの2点が問題となってくる。これに対して(3)は単一の立体以上のものを扱うことを目的としている。これは非多様体モデルが表現対象を3次元空間の胞体分割にまで拡張したことを示している。また(4)は、サーフェスモデルやワイヤフレームモデルを含んだ複合形状モデルを対象としたものである。一般に表現能力を拡張すればデータ構造は冗長になる。たとえば同じ多様体立体を表現しようとする場合、場合によって非多様体モデルの位相データ量は、多様体立体モデルの2~3倍

† 東京大学大学院総合文化研究科
Graduate School of Arts and Sciences, University of
Tokyo

程度になる。表現対象や処理対象に応じて適切な位相モデルを用いる必要があることはいうまでもないことである。本研究では「隣接位相要素」という概念によって、境界表現の位相モデルを分類し、それらの間の相連を明確にする。さらに各位相モデルのデータ構造、操作法などについて検討する。

2. 隣接位相要素による位相上の制約

本研究の前段階の研究として、3次元空間に埋め込み可能な一般の位相モデルを検討した⁸⁾。その際に、3次元空間内の点集合である0~3胞体を基本的な位相要素 Vertex, Edge, Face, Region とし、それら基本位相要素間の隣接関係を隣接位相要素によって表現した。したがって、隣接位相要素は表1に示すように、2つの基本的位相要素の組合せに応じて6種類定義される。図1は、これらの隣接位相要素を図で表したものである。

ところで境界表現における位相モデルにおいては、サイクルを中心とした順序情報が重要な役割を担っている。特に3次元空間に埋め込み可能な位相の場合には、図2に示すように、Disk (Vertex) サイクル, Radial (Edge) サイクル, Loop (Face) サイクルと呼ばれる3種類のサイクルが存在することが知られている^{4),9)}。隣接位相要素はこれらのサイクルを一意に表現するのに非常に有効である。つまり、Disk サイクルは Vertex 周りの end と fan のサイクル, Radial サイクルは Edge 周りの blade と wedge のサイクル,

Loop サイクルは Face 周囲の fan と blade のサイクルと見なすことができる。

2.1 境界における制約

本節では、各基本位相要素の境界における制約について簡単に述べる。議論の詳細については、既発表の論文¹⁰⁾を参照していただきたい。まず最初に Vertex は境界を持たないので議論の対象外である。次に Edge の境界は、その両端の end であり、1本の Edge は2つの end を持つ。したがって、Edge 境界では以下の制約式が得られる。

$$N_{\text{end}} = 2N_{\text{Edge}} \quad (1)$$

ただし、 N_x は位相要素 x の個数を表すものとする。

Face の境界である Loop 上では、図2(c)からも分かるように、fan と blade が交互に並んでサイクルをなしており、その個数は等しくなる。したがって、Face 境界では以下の制約式が得られる。

$$N_{\text{blade}} = N_{\text{fan}} \quad (2)$$

Region の境界である Shell の構成要素を corner, wedge, side と考えると、Shell は2-多様体と見なせる。したがって、Region の境界については、多様体立体的オイラー式¹¹⁾を拡張した以下の制約式が得られる。

$$(N_{\text{corner}} - N_{\text{a_disk}}) - N_{\text{wedge}} + (N_{\text{side}} - 2N_{\text{a_Loop}}) = 2(N_{\text{Shell}} - N_{\text{C_Cycle}}) \quad (3)$$

ここで a_Loop とは、Face に2つ以上の Loop がある場合の、2番目以降の Loop を意味している。平面上の Face の場合には、内部に存在する Loop と考えてもよい。 a_disk とは disk に関して a_Loop に相当するもので、1つの corner に2つ以上の disk がある場合の2番目以降の disk を表す。 C_Cycle は位相幾何でカットサイクルと呼ばれ、胞複体上のサイクルで1点に縮退しないもののことである。たとえば、球面上には0個、トーラス面上には2個の C_Cycle が存在する。

2.2 点の近傍における制約

次に、基本位相要素内の点の近傍における制約について考察する。図3は Face の近傍を図示したもので、

表1 隣接位相要素
Table 1 Coupling entities.

	Vertex	Edge	Face	Region
Vertex	***	end	fan	corner
Edge	end	***	blade	wedge
Face	fan	blade	***	side
Region	corner	wedge	side	***

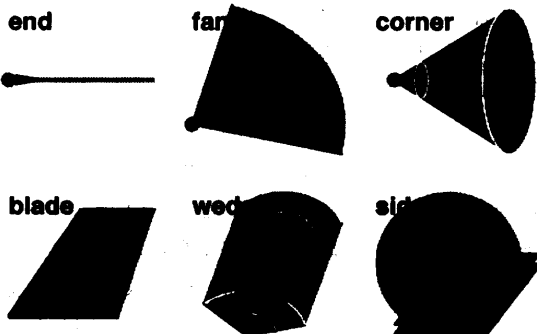


図1 隣接位相要素
Fig.1 Coupling entities.

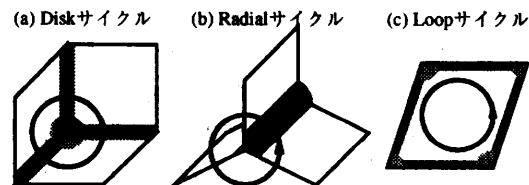


図2 3種類のサイクル
Fig.2 Three kinds of cycles.

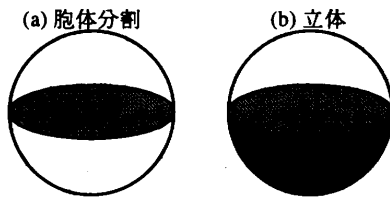


図3 Faceの近傍

Fig. 3 Neighborhoods of face points.

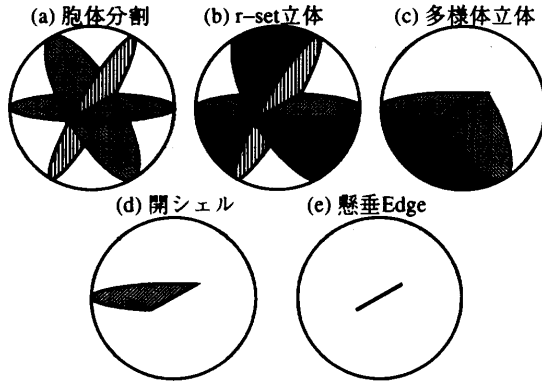


図4 Edgeの近傍

Fig. 4 Neighborhoods of edge points.

Faceとその上下の半球の3つの部分から構成される。この2つの半球はFaceとRegionの隣接部分に対応し、sideと見なせる。胞体分割の場合、各sideに隣接するRegionは、一般には異なるが、仮に同じであっても独立した2つのsideが存在するものとする。つまり、図3(a)のようにFaceに関してsideは対等となり、以下の式が成り立つ。

$$N_{side} = 2N_{Face} \quad (4)$$

一方、表現対象を立体とすると、すべてのFaceは向きづけ可能となる。つまり、sideの一方は立体内部、他方は立体外部に対応する。これを図示したものが図3(b)であり、式(4)に加えて、次式も成り立つ。

$$N_{i_side} = N_{o_side} \quad (5)$$

このように立体の場合、内部の空間と外部の空間を区別して考え、各空間の境界をそれぞれ内部Shellおよび外部Shellとする。以下、内部Shellに対応する位相要素 x は i_x 、外部Shellに対応するものを o_x と表すこととする。当然、 $N_x = N_{i_x} + N_{o_x}$ が成り立つ。

次に図4をもとにEdgeの近傍について考察する。胞体分割の場合は、(a)のようにEdgeの周囲にbladeとwedgeが交互に現れ、以下の式が成り立つ。

$$N_{blade} = N_{wedge} \quad (6)$$

r-set 立体の場合には、図4(b)のようにEdge周りに立体の内部と外部に対応するwedgeが交互に現れ

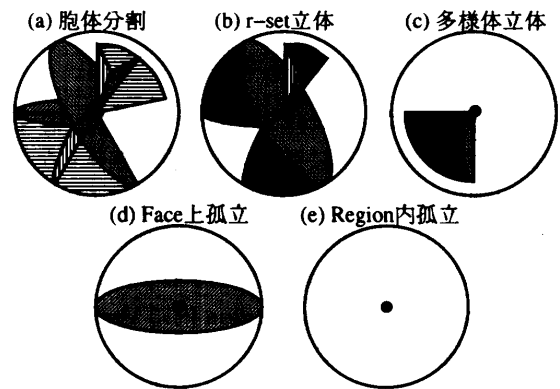


図5 Vertexの近傍

Fig. 5 Neighborhoods of vertex points.

る。したがって式(6)に加え、次式も成り立つことになる。

$$N_{i_wedge} = N_{o_wedge} \quad (7)$$

多様体立体の場合には、図4(c)のように立体内部と立体外部に対応するwedgeが各1つとなり、式(6)、(7)に加えて、次式も成り立つ。

$$N_{i_wedge} = N_{o_wedge} = N_{Edge} \quad (8)$$

上記の分類においては、いずれにおいても、式(6)は成立していた。サーフェスモデルやワイヤフレームモデルも考慮した複合形状モデルの場合には、どうなるであろうか。サーフェスモデルあるいは開シェルの端部では、図4(d)のようにEdgeは1つのFaceにしか隣接しないが、式(6)は成立する。一方、ワイヤフレームモデルに見られる懸垂Edgeは、図4(e)のようになる。このときにEdgeとRegionの隣接関係をwedgeとすると式(6)が成立しなくなる。したがって、このようなEdgeは胞体分割と同列に扱うことは不可能であり、懸垂EdgeとRegionの隣接関係はwedgeではなく、別の隣接関係として表現してはならない。

最後に図5をもとにVertexの近傍について考える。一般にVertexの近傍は(a)のようになるが、この近傍微小球表面上のグラフから次式が導ける⁸⁾。

$$N_{end} - N_{fan} + (N_{corner} - N_{a_disk}) = 2N_{Vertex} \quad (9)$$

r-set 立体の場合、図5(b)のように球面上のグラフが2色に塗り分けられる。これは通常の2次元グラフが4色に塗り分けられるのと異なっているが、この性質自体は、r-set 立体のEdgeに関する制約である式(7)から導ける。多様体立体の場合は、図5(c)のように立体の内部と外部に対応するcornerが各1つ存在し、a.diskは存在しない。したがって式(9)に加えて以下の式が成り立つこととなる。

$$\begin{aligned} N_{a_disk} &= 0 \\ N_{i_corner} &= N_{o_corner} = N_{Vertex} \end{aligned} \quad (10)$$

上記の分類では、Vertex 近傍はいずれも式 (9) を満たしていたが、球面上にグラフが定義できなくなると式 (9) が成立しなくなる。具体的には、図 5 (d) のように Face 上に Vertex が孤立している場合には、Vertex と Face の隣接関係を fan と定義すると、式 (9) が成立しなくなる。図 5 (e) は Region 内に Vertex が孤立している場合である。このときも Vertex と Region の隣接関係を corner とすると式 (9) が成立しなくなる。したがって、Face 上や Region 内に孤立した Vertex の隣接関係を表現するにあたっては、fan や corner として扱うべきではないことが分かる。

3. 位相モデルの分類

前章の議論から境界表現の位相モデルは次のように分類できる。

- 胞体分割：式 (1)～(4), (6), (9) 成立
複数の Region を対象とする位相モデル。Face を境とした 2 つの Region は対称に扱われ、サーフェスモデルなどの開シェルも扱える。隣接位相要素 feather を用いたデータ構造によって表現できる。
 - r-set 立体：式 (1)～(7), (9) 成立
体積を持つ単連結な点集合を対象とした位相モデル。すべての Face が整合的に向きづけられる。隣接位相要素 halfwedge によって表現できる。
 - 多様体立体：式 (1)～(10) 成立
境界が 2-多様体となる単一 Region を対象とした位相モデル。すべての Face が整合的に向きづけ可能で、各 Edge に 2 つの Face が隣接し、各 Vertex には 2 つの Region のみが隣接する。隣接位相要素 halfedge によって表現できる。
 - Face 内孤立 Vertex：式 (9) 不成立
Face 内の孤立 Vertex の表現には例外的な処理が必要となる。このような孤立 Vertex は r-set 立体や多様体立体とも共存する。
 - 懸垂 Edge：式 (6) 不成立
ワイヤフレームモデルに見られるような Region 内の懸垂 Edge の表現には、胞体分割や r-set 立体、多様体立体と別の表現が適当となる。
 - Region 内孤立 Vertex：式 (9) 不成立
Region 内の孤立 Vertex の表現には胞体分割や r-set 立体、多様体立体と別の表現が適当となる。
- 前半の 3 つの位相モデルは、最初の胞体分割を基本として、r-set 立体そして多様体立体と徐々に制約が厳

しくなっている。したがって、これらの位相モデルの間には、表現力に関して以下の包含関係が存在する。

胞体分割 ⊃ r-set 立体 ⊃ 多様体立体

これに対して後半の 3 つはそれぞれ独立した条件であり、Face 内孤立 Vertex、懸垂 Edge、Region 内孤立 Vertex の存在のみに依存する。これらの扱いについては、すでに提案した手法¹⁰⁾が有効と考えられる。そこで、以下では、前半の 3 つの位相モデルのデータ構造と操作について議論する。

4. データ構造

本章では、胞体分割、r-set 立体、多様体立体という 3 種類の位相モデルの計算機内データ構造について説明する。

4.1 胞体分割のための feather データ構造

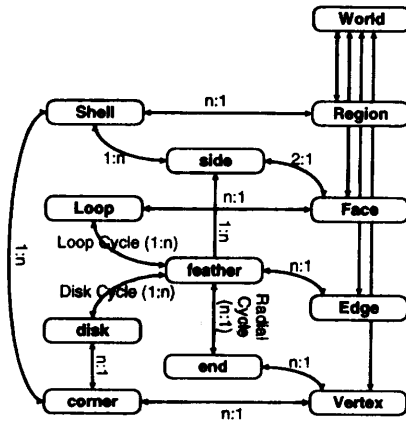
胞体分割を対象とする非多様体モデルとしては、図 6 (a) の feather を用いたデータ構造⁸⁾がある。式 (2) と式 (6) から、fan, blade, wedge の 3 つの位相要素の個数が等しく、それぞれに対応関係があることが分かる。そこで、これら 3 つの隣接位相要素を feather を用いて間接的に表現する。つまり、fan, blade, wedge と、これらによって構成される 3 つサイクルは、feather 間のポイントによって表される。まず、fan, blade, wedge は、2 つ 1 組の feather で表現されるので、これらの関係を表す FM (Fan Mate), BM (Blade Mate), WM (Wedge Mate) という 3 つのポイントが考えられる。さらに、3 種類のサイクルと、それをたどる順序から、CD (Clockwise Disk), CCD (Counter Clockwise Disk), CR (Clockwise Radial), CCR (Counter Clockwise Radial), CL (Clockwise Loop), CCL (Counter Clockwise Loop) という 6 つのポイントが考えられる。これらのポイントの間には、以下の 6 種類の関係が見いだせる。

$$\begin{aligned} x &= \text{CCD}(\text{CD}(x)) = \text{CD}(\text{CCD}(x)), \\ x &= \text{CCR}(\text{CR}(x)) = \text{CR}(\text{CCR}(x)), \\ x &= \text{CCL}(\text{CL}(x)) = \text{CL}(\text{CCL}(x)), \\ \text{CCD}(x) &= \text{CCR}(\text{FM}(x)), \\ \text{CCR}(x) &= \text{WM}(\text{BM}(x)), \\ \text{CCL}(x) &= \text{FM}(\text{BM}(x)). \end{aligned}$$

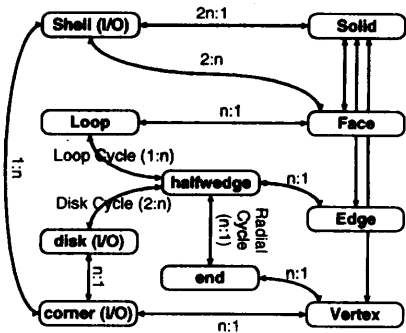
したがって、9 つのポイントの内の独立な 3 つのポイントを持てばよいことが分かる。3 つの基本ポイントとしては、効率や便利さなどの観点から、FM, BM, WM の組合せが適当と考えられる⁸⁾。

4.2 r-set 立体のための halfwedge データ構造

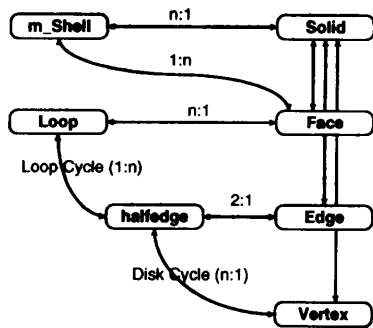
胞体分割と r-set 立体を比較すると、r-set 立体には



(a) feather データ構造



(b) halfwedge データ構造



(c) halfedge データ構造

図6 境界表現のためのデータ構造

Fig. 6 Data structures for boundary representations.

以下の性質が認められる。

- すべての Shell は立体の内部あるいは外部のいずれかを表している。
- Face の 2 つの side のうち、一方は立体の内部、他方は外部に対応する。
- Edge の周りでは内部と外部の 2 種類の wedge が交互に並ぶ。
- blade と fan には裏表の区別がある。

まず最初の性質は、r-set 立体において複数の Region を明示的に表現する必然性がないことを示している。2 番目の性質は、Face が 2 枚の side を並列に表現し

なくてよいことを示している。さらに、3 番目と 4 番目の性質は、fan, blade, wedge をわざわざ 2 つに分割して feather として表現せずに済むことを示している。そこで、feather に代わり、wedge の片側と blade, fan に対応する halfwedge を導入し、これを基本としたデータ構造を考える。図 6(b) は、このデータ構造を示したものである。

次に隣接順序関係を表現するための halfwedge 間のポインタについて考える。feather データ構造で考えた 9 種類のポインタをもとに議論を進める。まず、halfwedge は fan および blade には 1 対 1 に対応するものであるから、feather にあった FM と BM というポインタは不要となる。一方、wedge と disk に関しては、立体内部と立体外部の区別が必要になる。したがって、WM や CD, CCD という 3 種類のポインタは、立体の内部あるいは外部に応じて、さらに 2 種類に分けて扱わなくてはならない。つまり、WM ポインタの代わりに IM (Inside wedge Mate) と OM (Outside wedge Mate), CD と CCD というポインタの代わりに CI (Clockwise Inside disk) と CCI (Counter Clockwise Inside disk) および CO (Clockwise Outside disk) と CCO (Counter Clockwise Outside disk) というポインタが必要となる。これ以外に、CR, CCR, CL, CCL という 4 つのポインタがあるので、全部で 10 種類のポインタが考えられる。これらのポインタの間には、次の 7 つの関係式が成り立つ。

$$\begin{aligned}
 x &= CCI(CI(x)) = CI(CCI(x)), \\
 x &= CCO(CO(x)) = CO(CCO(x)), \\
 x &= CCR(CR(x)) = CR(CCR(x)), \\
 x &= CCL(CL(x)) = CL(CCL(x)), \\
 CCR(x) &= OM(IM(x)), \\
 CCL(x) &= CCI(IM(x)), \\
 CCL(x) &= CO(OM(x)).
 \end{aligned}$$

したがって、halfwedge データ構造の場合には、3 種類のポインタが独立となる。最少の組合せで他のポインタを簡潔に表現できるものとして、CCI, CCO, CL の組合せがある。また多少冗長ではあるが、すべてのポインタを非常に簡潔に表現できる組合せとして、IM, OM, CL, CCL の組合せも考えられる。

4.3 多様体立体のための halfedge データ構造

多様体立体と r-set 立体の比較においては、式 (8) と式 (10) に由来した以下の相違が認められる。

- 立体の内部と外部は完全に対称である。
- 各 Edge について内部 wedge と外部 wedge がそれぞれ 1 つだけ存在する。
- 各 Vertex の近傍には内部 corner と外部 corner

がそれぞれ1つ存在する。

- Vertexの近傍にa.diskは存在しない。

最初の性質からShellを内部と外部で独立に持つ必要がないことが分かる。2番目の性質は、多様体立体では1本のEdgeの回りにRadialサイクルと呼ぶほどのものが存在せず、1組のhalfwedge, すなわちhalfedge, で十分であることを示している。3番目の性質はcornerが不要であること、最後の性質はdiskが不要であることをそれぞれ示している。そこで、内部Shellと外部Shellを共通に表現することとし、これをm.Shellと名づける。さらに隣接位相要素として、cornerとdiskをなくした、halfedgeデータ構造を考える。図6(c)は、このデータ構造を示したものである。

halfedgeにはどのようなポイントが必要であろうか。まず、Radialサイクルが存在せず、単にhalfedgeのペアがあるにすぎないで、CR, CCRというRadialサイクルのポイントはなくなり、IM, OMというポイントはEM (Edge Mate) というポイントに統合できる。また、Vertex周りにはcornerが2つしか存在せず、それらが完全に対称であることから、Diskサイクルも1つだけ表現すればよい。したがって、CI, CCI, CO, CCOというポイントは統合されて、CD (Clockwise Disk), CCD (Counter Clockwise Disk) を用いる。これ以外に、CL, CCLという2つのポイントが残るので、全部で5種類のポイントが考えられる。これらのポイントの間には、次の3つの関係式が成り立つ。

$$\begin{aligned}x &= \text{CCL}(\text{CL}(x)) = \text{CL}(\text{CCL}(x)), \\x &= \text{CCD}(\text{CD}(x)) = \text{CD}(\text{CCD}(x)), \\ \text{CCL}(x) &= \text{CD}(\text{EM}(x)).\end{aligned}$$

したがって、halfedgeデータ構造の場合には、2種類のポイントが独立となる。最少の組合せで他のポイントを簡潔に表現できるものとして、CD, CLの組合せがある。また多少冗長ではあるが、EM, CL, CCLの組合せも、他のポイントを簡潔に表現できる。このポイントの組合せはMäntyläのhalfedgeデータ構造にはかならない¹⁾。

5. 位相操作

2章で議論した制約式は、位相操作の定義にも利用できる。境界表現モデルにおいて位相上の制約を考慮した操作として、従来よりオイラー操作が知られている^{1), 12)}。ここでは各位相モデルについて、オイラー操作およびオイラー操作よりもさらに低レベルの隣接位相操作について考える。

5.1 胞体分割モデルのオイラー操作と隣接位相操作

オイラー操作とは、オイラー式、あるいはEuler-

Poincaré式、を保存する操作である。ここで重要なのは各々のモデルに沿ったオイラー式の解釈を得ることにある。オイラー式は一般に胞体の個数とベッチ数の関係式と考えられる。そこで、(1)~(4), (6), (9)の6式から、隣接位相要素に関する項を消去し、基本位相要素すなわち胞体に関する制約式を求める。

$$\begin{aligned}N_{\text{Vertex}} - N_{\text{Edge}} + (N_{\text{Face}} - N_{\text{a_Loop}}) \\ = N_{\text{a_Shell}} - N_{\text{C_Cycle}} + N_{\text{Region}}\end{aligned}\quad (11)$$

式(11)を保存する操作群を胞体分割モデルのオイラー操作と定義できる。

featherデータ構造には、胞体そのものを表す位相要素以外にも、featherを初めとした隣接位相要素が複数存在する。つまり、位相操作を実現するにあたっては、隣接位相要素の生成消去に関する基準も必要となる。この隣接位相要素の生成消去の基準として、点の近傍における制約式である式(4), (6), (9)が利用できる。これらの式を保存する操作を隣接位相操作と名付け、この隣接位相操作の組合せによってオイラー操作を実現すればよい。胞体分割モデルのオイラー操作ならびに隣接位相操作の詳細はすでに発表した論文⁸⁾を参照されたい。

5.2 r-set 立体のオイラー操作と隣接位相操作

r-set 立体のオイラー操作は、式(11)をもとに導くことができる。halfwedgeデータ構造にはRegionという位相要素がないため、 N_{Region} という項は不適当である。ところで、Shellの総数はRegionとa.Shellの数の合計に等しく、 $N_{\text{Shell}} = N_{\text{Region}} + N_{\text{a_Shell}}$ という式が成り立つ。これを式(11)に代入することによって、以下のr-set 立体のためのオイラー式が得られる。

$$\begin{aligned}N_{\text{Vertex}} - N_{\text{Edge}} + (N_{\text{Face}} - N_{\text{a_Loop}}) \\ = N_{\text{Shell}} - N_{\text{C_Cycle}}\end{aligned}\quad (12)$$

式(12)を保存する操作がr-set 立体のためのオイラー操作である。基本的なオイラー操作の組合せとしては、図7に示すmVFSS (make Vertex Face Shell), mEV (make Edge Vertex), mEF (make Edge Face), mEkL (make Edge kill a_Loop), mVkc (make Vertex kill C_Cycle), mEC (make Edge C_Cycle), mVS (make Vertex Shell), mEkS (make Edge kill Shell)の8つの操作とそれらの逆操作が考えられる。他のオイラー操作はこれらの基本オイラー操作を結合することによって実現できる。

halfwedgeデータ構造においても多く隣接位相要素がある。したがって、隣接位相要素を処理するための、隣接位相操作が不可欠となる。r-set 立体の隣接位相操作は、式(4)~(7), (9)を満たす。図8は前に示した8

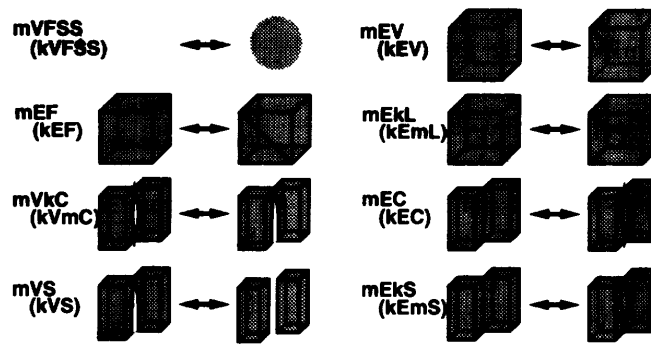


図7 r-set 立体のオイラー操作
Fig. 7 Euler operations for r-set solids.

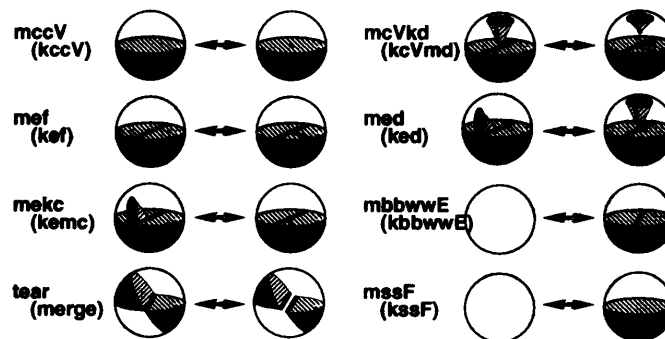


図8 r-set 立体の隣接位相操作
Fig. 8 Neighborhood operations for r-set solids.

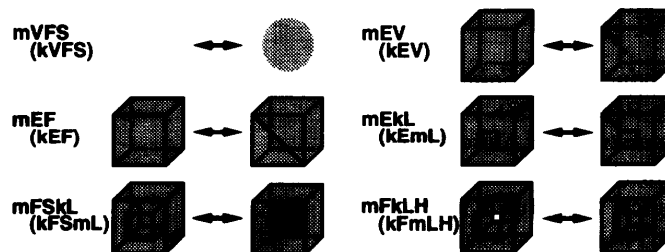


図9 多様体立体のオイラー操作
Fig. 9 Euler operations for manifold solids.

つのオイラー操作を実現するために必要となる隣接位相操作を示したものである。mccV (make corner corner Vertex), mcVkd (make corner Vertex kill a disk), mef (make end fan), med (make end a disk), mekc (make end kill corner) の5つの操作はVertexの近傍に関するものであり、mbbwE (make blade blade wedge wedge Edge) とtearの2つの操作はEdgeの近傍、mssF (make side side Face) はFaceの近傍に関する操作である。

5.3 多様体立体のオイラー操作と隣接位相操作

多様体立体のオイラー式は、r-set 立体との相違から、式(12)を変形することによって導ける。多様体立

体の場合には、m_Shellによって立体内部のShellと外部のShellを表しているから、 $N_{\text{Shell}} = 2N_{\text{m_Shell}}$ という関係が成り立つ。また、多様体立体のHandleは2つのC_Cycleに対応する。したがって、 $N_{\text{C_Cycle}} = 2N_{\text{Handle}}$ という関係が成り立つ。これらの関係を式(12)に代入して次の多様体立体のオイラー式が得られる。

$$N_{\text{Vertex}} - N_{\text{Edge}} + (N_{\text{Face}} - N_{\text{a_Loop}}) = 2(N_{\text{m_Shell}} - N_{\text{Handle}}) \quad (13)$$

この式は多様体立体のオイラー式として知られている式¹¹⁾にはかならない。多様体立体の基本オイラー操作としては、図9に示す6つの操作、mVFS (make

Vertex Face m_Shell), mEV (make Edge Vertex), mEF (make Edge Face), mEkL (make Edge kill a_Loop), mFSkL (make Face m_Shell kill a_Loop), mFkLH (make Face kill a_Loop Handle) とこれらの逆操作がよく知られている。また halfedge データ構造では、隣接位相要素と呼べるものは halfedge しかない。この halfedge も 1 つの Edge について 2 つ存在するという制約に従うのみである。したがって、隣接位相操作の必要性はないといってよい。

6. おわりに

本研究では境界表現モデルの位相を隣接位相要素の観点から分類し、胞体分割, r-set 立体, 多様体立体という 3 種類の位相モデル間の相違について考察した。次に、これらの位相モデルを表現するためのデータ構造を示した。最後に各位相モデルの生成消去を実行するオイラー操作と隣接位相操作について述べた。それぞれの位相モデルは、各々表現対象が異なっており、アプリケーションに応じて使い分けことが望まれる。以上より、本研究では、胞体分割を基本とした 3 つの位相モデルの計算機内実現法について、一般的な解釈を与えられたと考えられる。この解釈は境界表現を基礎とする形状モデル間のデータ交換の際に必要な不可欠な概念といえる。

参 考 文 献

- 1) Mantyla, M.: *An Introduction to Solid Modeling*, Computer Science Press (1988).
- 2) Hoffmann, C.M.: *Geometric and Solid Modeling*, Morgan Kaufmann (1989).
- 3) Weiler, K.J.: *Topological Structures for Geometric Modeling*, PhD Thesis, Rensselaer Polytechnic Institute (1986).
- 4) Gursoz, E.L., Choi, Y. and Prinz, F.B.: Vertex-based Representation of Non-manifold Boundaries, *Geometric Modeling for Product Engineering*, Wozny, M.J., Turner, J.U. and Preiss, K. (Eds.), pp.107-130, North-Holland (1990).
- 5) Desaulniers, H. and Stewart, N.F.: An Extension of Manifold Boundary Representations to the r-Sets, *ACM Trans. Graphics*, Vol.11, No.1, pp.40-60 (1992).
- 6) Higashi, M., Yatomi, H., Mizutani, Y. and Murabata, S.: Unified Geometric Modeling by Non-manifold Shell Operation, *Proc. Second Symposium on Solid Modeling and Applications*, Rossignac, J., Turner, J. and Allen, G. (Eds.), pp.75-84, A.C.M. (1993).
- 7) Requicha, A.A.G.: Representations for Rigid Solids: Theory, Methods, and Systems, *Computing Surveys*, Vol.12, No.4, pp.437-464 (1980).
- 8) 山口 泰, 木村文彦: 非多様体位相の隣接関係の表現と操作, 情報処理学会論文誌, Vol.32, No.6, pp.731-739 (1991).
- 9) Brisson, E.: Representing Geometric Structures in d Dimensions: Topology and Order, *5th Symposium on Computational Geometry*, pp.218-227, A.C.M. (1989).
- 10) Yamaguchi, Y. and Kimura, F.: Representation and Operations for Non-manifold Topology Based on Coupling Entities, *Computer Graphics and Applications*, Vol.15, No.1, pp.42-50 (1995).
- 11) Braid, I.C., Hillyard, R.C. and Stroud, I.A.: Stepwise Construction of Polyhedra in Geometric Modelling, *Mathematical Method in Computer Graphics and Design*, pp.123-141, Academic Press (1980).
- 12) Baumgart, B.: Winged-edge Polyhedron Representation, Technical Report, Stanford University, A.I. Report, No.CS-320 (1972).

(平成 8 年 6 月 14 日受付)

(平成 8 年 10 月 1 日採録)



山口 泰 (正会員)

1961 年生。1988 年東京大学大学院工学系研究科情報工学専攻博士課程修了。同年東京大学教養学部図学教室助手。1989 年東京電機大学工学部講師。1993 年より東京大学大学院総合文化研究科助教授。この間、1995～1996 年スタンフォード大学客員研究員。形状モデリング、機械系 CAD/CAM、コンピュータグラフィックスなどの研究に従事。工学博士。精密工学会、ACM、IEEE などの会員。