

## 半空間プリミティブに基づく多面体表現のための 多角形パッチ生成における冗長性削除の一手法

河合善之<sup>†,☆</sup> 岡田 稔<sup>††,☆☆</sup>

本論文では CSG (Constructive Solid Geometry) で定義された物体表現からポリゴン表現への一変換手法について述べる。CSG において多面体をポリゴンに変換する際、集合演算によって生成される幾何学的に冗長な頂点や稜線の存在が問題となる。この冗長な頂点や稜線を削除し、連結した複数のポリゴンを 1 つのポリゴンに統合することによって、冗長性のないデータ列を生成することを目標とした。この変換手法と最適凸多角形分割手法を組み合わせることにより、効率の良いレンダリングが可能となる。すなわち、ポリゴンパッチ (サーフェスモデル) をレンダリングする際はもちろん、レイ・トレーシングやビーム・トレーシングなど、CSG 表現をポリゴン変換するモデリングに適用すれば、レンダリングの高速化が期待される。本研究では、半空間プリミティブによる CSG を VRML ソースプログラムに変換する問題に対して提案手法を適用し、本手法の有効性を確認した。

### A Method to Eliminate Redundancy in Polygonal Patch Generation to Express Polyhedron Based on Half-space Primitives

YOSHIYUKI KAWAI<sup>†,☆</sup> and MINORU OKADA<sup>††,☆☆</sup>

In this paper, we propose a method to transform solid expressions, defined by CSG (Constructive Solid Geometry), into polygonal expressions. When transforming a CSG object into polygons, the existence of geometrically redundant vertices and edges, generated by the set operations, becomes problematic. By eliminating these redundant vertices and edges, and unifying the connected polygons into a single polygon, we aim to generate the polygon data without any redundancy. The combination of this transforming method and the method of dividing the best suited convex polygon makes it possible to achieve an efficient rendering. In other words, if applied to other modeling methods which transform CSG expressions into polygonal expressions, such as ray-tracing or beam-tracing, and of course polygon-patch (surface model) rendering, a high speed rendering would be expected. We applied the proposed method in this research to the issue of outputting the VRML source program when inputting CSG in Half-space primitives and have confirmed its effectiveness.

#### 1. はじめに

CG (Computer Graphics) において、現在用いられている代表的なソリッドモデリングの形状記述方式として、CSG (Constructive Solid Geometry), B-reps

(Boundary representations) などがあげられる。

CSG 表現では単純な形状 (プリミティブ) を集合演算子により結合し、より複雑な物体を生成することができる。これは人間による直観的形狀把握と構成が容易であるといえる。一方、B-reps はサーフェスモデルを拡張して物体の内部と外部の区別をすることができるようにしたデータ構造である。しかし、そのデータ構造は複雑であり、人間が B-reps データを入力するのは非常に困難である。

また、近年、数十万ポリゴン/秒の描画が可能な多面体描画エンジン (以下、ポリゴンエンジン) を搭載したワークステーション (WS) など、ハードウェアの発達により多面体の高速描画が可能となってきたが、これらのシステムの多くはポリゴンデータを入力とし

<sup>†</sup> 名古屋大学大学院工学研究科電子情報学専攻  
Department of Information Electronics, Graduate  
School of Engineering, Nagoya University

<sup>☆</sup> 現在、株式会社 NTT コミュニケーションウェア  
Presently with NTT Communicationware Co.

<sup>††</sup> 名古屋大学大学院工学研究科情報工学専攻  
Department of Information Engineering, Graduate  
School of Engineering, Nagoya University

<sup>☆☆</sup> 現在、中部大学工学部工業物理学科  
Presently with Department of Engineering Physics,  
College of Engineering, Chubu University

て要求している。このため、CSG や B-reps などの表現をポリゴン表現に変換することが必要である<sup>1)</sup>。ところが、CSG で定義された物体をポリゴンに変換すると、一般的には幾何学的に冗長な頂点や稜線が数多く生成されてしまう。このため、ポリゴン数が増加し、レンダリングにおける計算量が増大する。

筆者らは先に、半空間に基づく CSG により効果的に多面体定義を行うことができることを示した<sup>3),4)</sup>。しかし、同方法により定義された多面体をポリゴンエンジン搭載 WS で描画するためにポリゴン変換を行うと冗長な頂点・稜線が数多く生成されてしまうという問題点が浮上した。そこで本論文では、CSG 表現からポリゴンパッチ表現に変換する過程で生じる冗長な頂点や稜線を削除する手法を提案する。

なお、多くのポリゴンエンジン搭載 WS では凸多角形の入力を要求している。本方法と三角形分割<sup>9)</sup>に代表される凸多角形分割法<sup>10)</sup>を組み合わせることにより、計算時間の削減、レンダリングの高速化などが期待される。もちろん、本方法と凸多角形分割の適用後では、再び冗長なポリゴンを生じるが、それはたかだか、凸多角形分割だけを適用した後のポリゴン数に等しい。一般には本方法により冗長な頂点・稜線を削除した後に凸多角形分割した場合に有効性が期待される。

## 2. 冗長な頂点・稜線の削除

### 2.1 冗長な頂点・稜線とは？

CSG において単純なプリミティブ同士の和演算 ( $A \cup B$ ) によって生成される冗長な頂点・稜線の例を図 1 に示す。図 1 において○は冗長な頂点の例であり、太線は冗長な稜線の例である。

図 1 の例は立方体どうしの和演算であるが、このような単純な演算をするだけでも 4 つの冗長頂点と、4 本の冗長稜線が生成されてしまう。これらの幾何学的に冗長な頂点・稜線は、レンダリングの結果 (ワイヤフレームなどを除く) には反映されないが計算量を増加させる原因となる。

### 2.2 CSG における物体定義

本論文ではプリミティブとして  $n$  個の半空間  $f_i(x, y, z) \leq 0$  を用いた。ここで、平面方程式  $S_i$  を以下のように定義する。

$$S_i : f_i(x, y, z) = a_i x + b_i y + c_i z - d_i = 0 \quad (1) \\ (0 \leq d_i, 0 \leq i < n)$$

また、プリミティブどうしの論理演算として 3 値論理演算<sup>3),4)</sup>を利用し、CSG データは文献 3), 4) 同様、Lisp などに用いられる S 式を準用して記述されている。

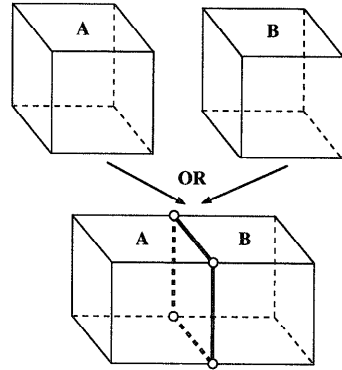


図 1 和演算によって冗長な頂点・稜線が生成される一例  
Fig. 1 An example of redundant vertices and edges in "OR" set operation.

### 2.3 頂点・稜線の選択方法

物体定義された CSG データの  $n$  平面データから頂点を求める。3 つの平面  $S_i, S_j, S_k$  をすべての組合せ ( ${}_n C_3$ ) で選択し、式 (2) により交点座標を算出する。

$$(x, y, z)^t = \mathbf{A}^{-1}(d_i, d_j, d_k)^t \quad (2)$$

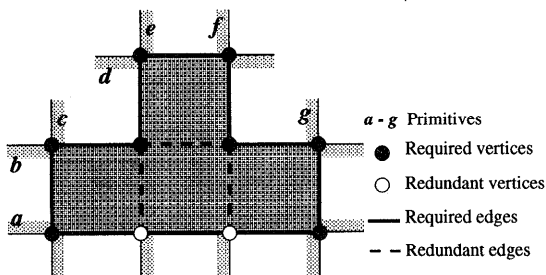
$$\mathbf{A} = \begin{bmatrix} a_i & b_i & c_i \\ a_j & b_j & c_j \\ a_k & b_k & c_k \end{bmatrix}$$

ただし、 $|\mathbf{A}| = 0$  のとき、交点なしとする。これにより求められた交点を物体定義で用いた S 式により 3 値論理評価<sup>3),4)</sup>を行う。3 値論理評価の結果、交点は領域の内部 (状態値 I)、領域の境界部 (状態値 B)、領域の外部 (状態値 O) のいずれかに分類される。その評価結果として状態値 B の交点をポリゴンの頂点候補点とする。ここで、頂点候補点とは冗長頂点 (偽の頂点) を含む頂点集合である。

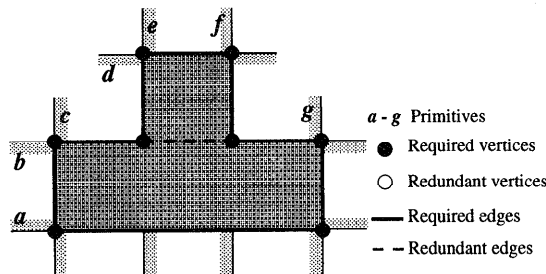
さて、稜線は 2 平面の交差によって生成される。注目平面とある 1 平面を考え、この 2 平面によって生成される稜線上では、この 2 平面以外の平面と、この稜線とによって頂点が生成される。これらの頂点間の線分のうち、その中点の 3 値論理評価が "B" の直線を稜線候補とする。

### 2.4 冗長な頂点・稜線

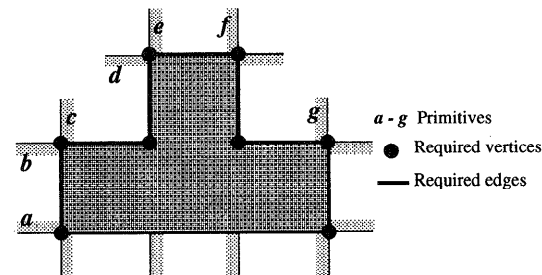
CSG において冗長な頂点・稜線が生成される例を図 2(a) に示す。図 2(a) は、多面体を構成する、ある 1 ポリゴンを垂直方向から見た図である。なお、○、●はともに 3 値論理評価によって状態値 B となった交点である。このため、従来の手法<sup>5)</sup>において○はポリゴンを構成するために必要な頂点と見なされてしまう。また、偽の稜線も同様に、ポリゴンを構成するために必要な稜線と見なされ、問題となる。以下に、こ



(a) Original status.



(b) Result of performing Step 1 to eliminate redundant vertices.



(c) Result of performing Step 2 to eliminating redundant edges.

図2 CSG表現： $(and (and a c d g) (or b (and e f)))$  によって生成される冗長性とその削除の例  
 Fig.2 Example of Redundancy and its elimination of  $(and (and a c d g) (or b (and e f)))$ .

れら冗長な頂点と稜線を削除する方法を説明する。

## 2.5 冗長な頂点・稜線の削除法

### 2.5.1 アルゴリズムの流れ

本手法では、冗長な頂点・稜線の削除を2 Stepで処理する。

Step 1では幾何学的に冗長な頂点をすべて削除する。図2(a)の物体から冗長な頂点を削除した結果を図2(b)に示す。なお、冗長稜線は2種類に大別することができる。すなわち、

- (1) 冗長頂点を少なくとも1つ持つ
- (2) 2つの冗長でない真の頂点を持つ

である。(1)のタイプの冗長稜線はすべての冗長稜線の大部分を占めることが実験的に明らかとなっている。冗長な頂点を削除する副作用として、(1)の稜線は自

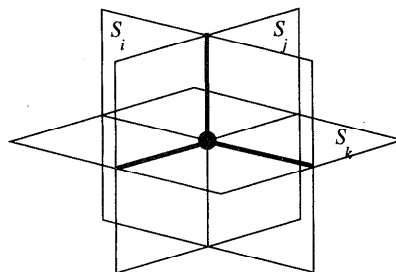


図3 3平面による頂点

Fig. 3 A vertex generated by 3 planes.

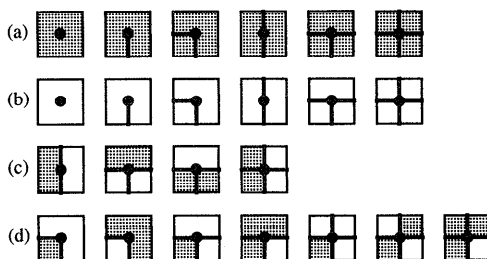


図4 3平面によって生成される23頂点パターン

Fig. 4 23 vertex patterns generated by 3 planes.

(a), (b), (c): redundant patterns, (d): significant patterns.

動的に削除できる。この一方では、(2)の冗長稜線が削除されないため、Step 2として、残りの冗長稜線を削除する。その結果を図2(c)に示す。

これら2つのStepにより、幾何学的に正しいポリゴンが生成できる。以下に、各Stepの手順を示す。

### 2.5.2 Step 1: 冗長頂点の削除

はじめに冗長な頂点をすべて削除する。多面体の頂点は少なくとも3平面により構成され、図3はその様子を示している。ここでその1つの面に注目すると4つの領域に分割される(各領域を四半領域という)。3平面によって生成される頂点パターンを図4に示す。ここで頂点パターンとは、多面体を構成する多角形上に存在する頂点とそれを含む近傍の状態を表し、ある平面上の頂点候補点を垂直方向から見たものである。[グレー]領域は物体の境界面(状態値Bの領域)、[白]領域は物体の内部または外部(状態値IまたはOの領域)を表している。

領域は、その内点を3値論理評価することにより、その3値論理評価を行う。本研究では、プリミティブに半空間プリミティブを用いたため、冗長性削除前ではすべてのポリゴンパッチは凸である。そのため、3値論理評価を行う内点を凸ポリゴンの重心とした。冗長頂点削除後でも、もとの凸ポリゴンの1つの重心を再利用する。

このとき、冗長頂点は以下に示す性質を呈する。

#### 【冗長頂点の性質】

- すべての領域に物体境界面が存在する (図 4 (a)).
- どの領域にも物体境界面が存在しない (図 4 (b)).
- グレー領域を融合した領域と白領域を融合した領域に角がない (図 4 (c)).

図 4 は付録で示すすべての頂点パターン 82 種のうち、回転対称性、軸対称性を持つ頂点パターンを除いた代表頂点パターン 23 種である。これらの代表頂点パターンのうち、幾何学的に必要な頂点パターンは、(d)であり、(a), (b), (c)の頂点パターンの頂点は幾何学的に冗長である。4以上の平面によって生成される頂点候補点についても同様に頂点パターン (a), (b), (c)を持つ頂点を削除すればよい。そこで、以下のような手順を構成する。

#### 【冗長頂点削除手順】

- (1) 多面体を構成する多角形を含むすべての平面について処理 (2) を行う。
- (2) 平面上のすべての頂点について処理 (3) を行う。
- (3) 図 4 (a), (b), (c) にあてはまる頂点をすべて削除する。 ■

この手順により、すべての冗長な頂点を削除できる。また、これにより大半の冗長な稜線も削除される。

#### 2.5.3 Step 2: 冗長稜線の削除

次に残りの冗長な稜線を以下に示す稜線追跡型手順によりすべて削除する。

ここで、ポリゴン上の真の稜線、および稜線の追跡方向を次のように定義する。

#### 【定義 1】 (真の稜線)

状態値 **B** の領域と状態値 **I**, **O** の領域との境界にある稜線を真の稜線という。 ■

定義 1 を満たさない稜線は冗長である。

#### 【定義 2】 (追跡方向となる稜線)

追跡方向となる稜線とは、基準となる頂点から見て、左側にはつねに状態値 **B** の物体の表面が存在しているような稜線である。 ■

図 2 (b) において残りの冗長な稜線を削除する過程を図 5 に示す。以下の手順において、CP は注目頂点、CE は注目稜線である。

#### 【冗長稜線削除手順】

- (1) すべての平面  $S_i$  に対して処理 (2)~(7) を繰り返す。
- (2) 平面  $S_i$  上のすべてのポリゴンが生成されるまで処理 (3)~(7) を繰り返す。
- (3) スタート点となる頂点を任意に決定し、CP とする。

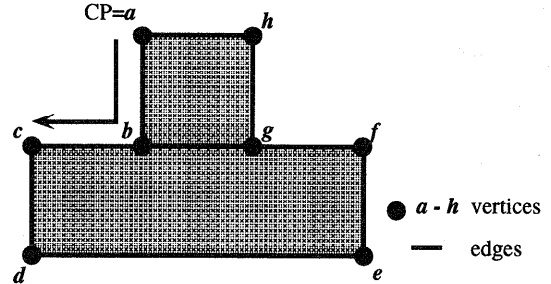


図 5 稜線追跡型手順による稜線削除

Fig. 5 Edge elimination by edge tracing algorithm.

- (4) CP に接続している稜線から定義 1 を満たさない稜線を削除する。
- (5) CP に接続している稜線から定義 2 を満たす稜線を選別し、稜線候補とする。
- (6) 次の基準により次の CP を選択する (CP=**a**).
  - 次の稜線候補が 1 本のみ存在する場合: その稜線を、CE とする。
  - 次の稜線候補が 2 本以上存在する場合: 定義 2 により、CP を中心として CE から時計回りの順に稜線の状態を調べ、初めて定義 1 を満たす稜線候補を CE とする。
- (7) スタート点に到達するまで処理 (4)~(6) を繰り返す。 ■

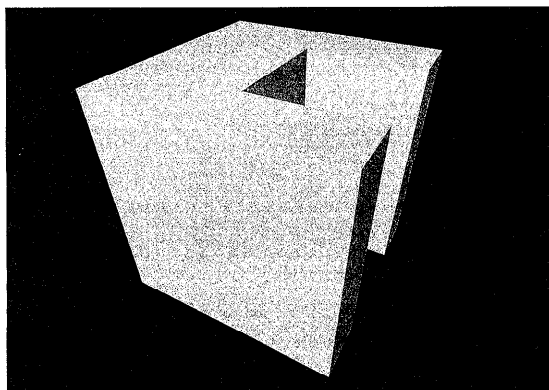
この手順では、稜線を追跡する際、Step 1 によって選択された真の頂点に接続している稜線の状態 (真偽) をすべて調べながら冗長な稜線をすべて削除している。

### 3. 実験

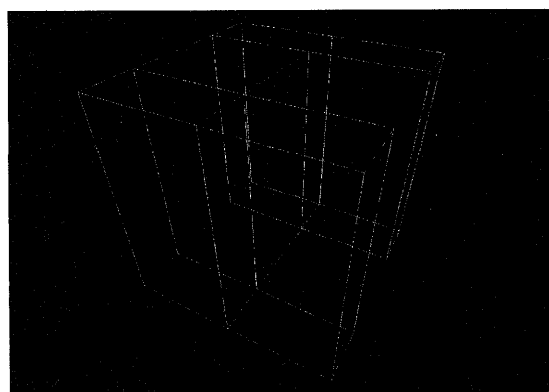
本手法の有効性を確認するために、冗長な頂点・稜線の削除をとまなうポリゴン変換方法と削除をとまなわない変換方法の結果を比較した。実験は半空間プリミティブを用いて物体を S 式によって定義した CSG データ<sup>3),4)</sup>を入力として VRML ソースリストを出力させることによって行った。

実験に用いた形状として物体 A (10 面体: プリミティブ数 9), B (28 面体: プリミティブ数 28), C (794 面体: プリミティブ数 70) の画像生成例をそれぞれ図 6, 図 7, 図 8 に示す。それぞれのサーフェスモデルによる対象物体の画像を各 (a) に、また、各物体の冗長な頂点・稜線について削除前、削除後の比較を各図 (b), (c) に示した。頂点・稜線の確認のためにワイヤフレーム表示を行っている。

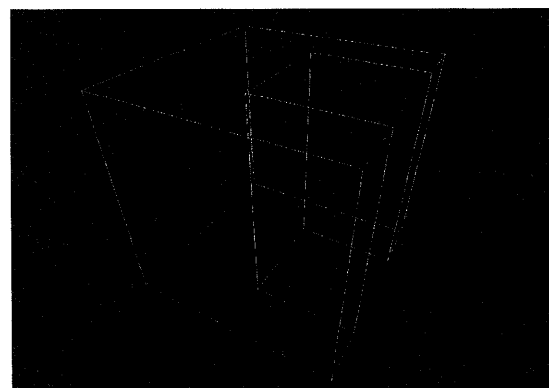
表 1 に示された頂点数、稜線数およびポリゴン数の削除前、削除後の変化からそれぞれ大幅に削減された



(a) Rendered image.



(b) Before elimination process.



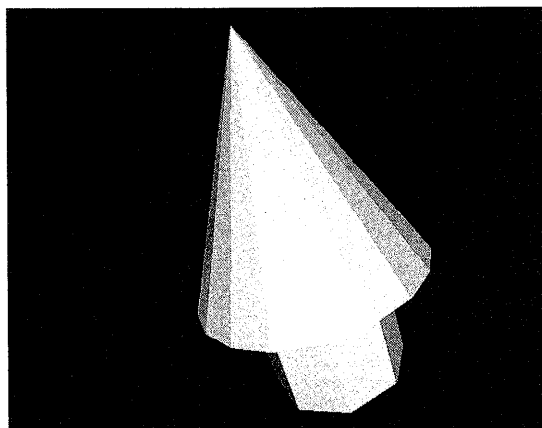
(c) After elimination process.

図6 物体A:凹の字多面体

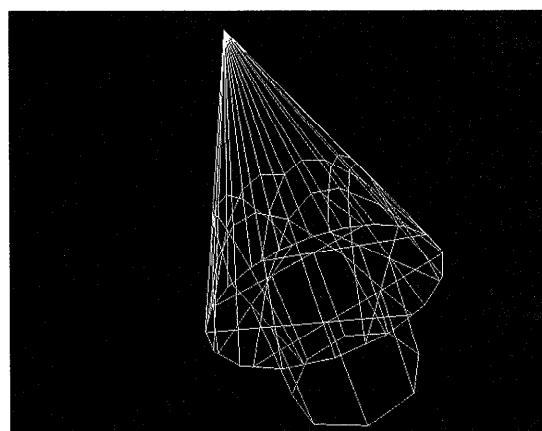
Fig. 6 Object A: a concave polyhedron.

ことが分かる.

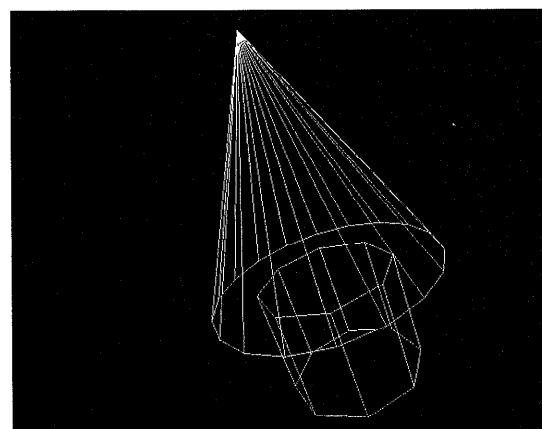
表2には, CSGデータを読み取り, VRMLのソースプログラムとして書き出すまでの処理を冗長な頂点・稜線を削除しない方法, 削除する方法を実行し, それぞれCPU時間を測定した結果を示す. また, 3値論理評価とポリゴン作成, 削除する方法でのStep 1, Step 2の時間の内訳もあわせて示した. この結果,



(a) Rendered image.



(b) Before elimination process.

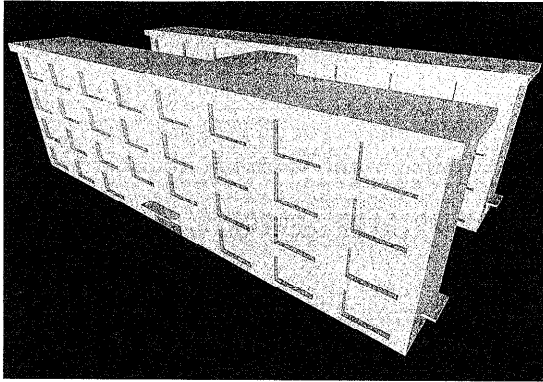


(c) After elimination process.

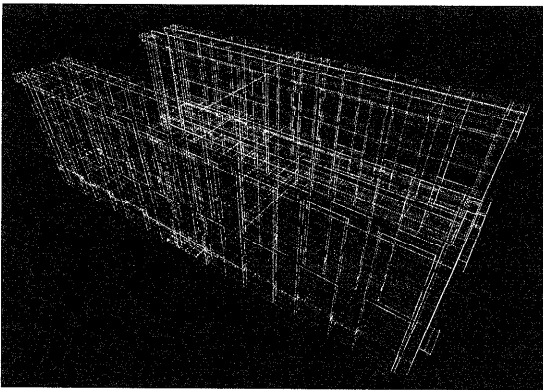
図7 物体B:曲面の多面体近似

Fig. 7 Object B: a curved surface by a polyhedron.

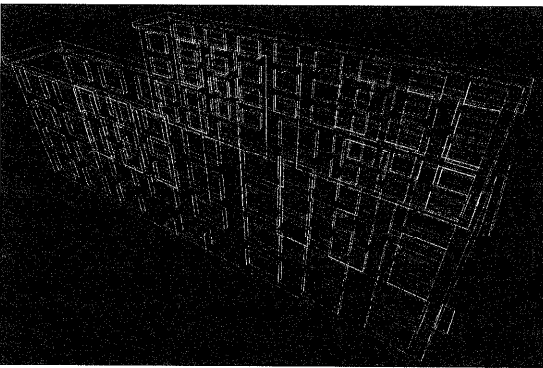
特にStep 2での計算量はStep 1に対して非常に小さく, また, 本方法の有無(冗長性削除処理の有無)による総計算量に大きな違いがないこと, ポリゴン作成にかかる計算量が劇的に削減されたことが分かる.



(a) Rendered image.



(b) Before elimination process.



(c) After elimination process.

図8 物体C:ある建物

Fig. 8 Object C: a building.

物体Cにおいて実時間、CPU時間ともかなりの時間がかかっているが、これは、物体定義に用いたS式が複雑であり、3値論理評価とStep 1に時間の大半を費やしていることが判明している。

以上の実験には Silicon Graphics 社の Origin 200 (R10000, 180 MHz) を用いた。

表1 頂点数, 稜線数, ポリゴン数の変化

Table 1 Change of vertices, edges and polygons.

		削除前	削除後	削除率
物体 A (10 面体)	頂点数	24	16	33%
	稜線数	44	24	45%
	ポリゴン数	22	10	55%
物体 B (28 面体)	頂点数	115	35	70%
	稜線数	230	56	76%
	ポリゴン数	114	28	75%
物体 C (794 面体)	頂点数	3582	958	73%
	稜線数	7164	1618	77%
	ポリゴン数	4384	794	82%

#### 4. ま と め

本論文では、多面体表現のための効率的なポリゴンパッチ生成の一手法として幾何学的に冗長な頂点および冗長な稜線を削除する手法を提案した。本手法により、CSGで定義された多面体の頂点数、稜線数を幾何学的に最小にした。また、実験においてはポリゴン数を10面体で55%、28面体で75%、794面体で82%削除でき、計算量を大幅に増加させることなく、ポリゴン数の最適化をともなったパッチ生成を行えることが確認された。

CSGにおいて生成されるポリゴンが最適となるようにユーザが意識しながらモデリングを行うことは非現実的であり、生成される物体形状がユーザの意図するものであるかどうかに関心してモデリング操作することが重要である。このような場合において本方法で示したような表現方式の自動変換は重要な問題である。本研究では、特にVRMLソースへの変換問題を中心として扱ったが、同様にCSGからポリゴンへの変換が必要となる手法(ポリゴンパッチ変換、ビーム・トレーシング<sup>6),11)</sup> etc.)に、本手法を適用すればレンダリングの高速化が期待できる。特に複雑な形状の物体をレンダリングする際などに威力を発揮すると考えられる。

本手法はいったん適用されると、形状が変化しない限り有効性が継続するため、仮想空間のウォークスルーなどにおいて有効である。なお、本手法の計算量は、プリミティブ数  $n$  に対し  $O(n^3)$  である。このため実験結果からも分かるようにプリミティブが増加するにつれて計算時間が急激に増加する。そのため、今後の課題として計算量を削減するために構成手順の改良が必要となる。また、曲面物体への対応、最適な凸多角形分割などがあげられる。

表2 各方法における計算時間とその主要内訳 [sec]  
Table 2 Comparison of calculation time in [sec].

	冗長性削除	総計算時間	3値論理評価	Step 1	Step 2	ポリゴン作成
物体 A (10 面体)	しない	0.004	0.00143	-	-	0.00108
	する	0.005	0.00143	0.001	0.0002	0.000135
物体 B (28 面体)	しない	0.077	0.0440	-	-	0.0149
	する	0.122	0.0854	0.015	0.001	0.000122
物体 C (794 面体)	しない	16.671	7.652	-	-	4.318
	する	16.543	8.321	5.406	0.338	0.215

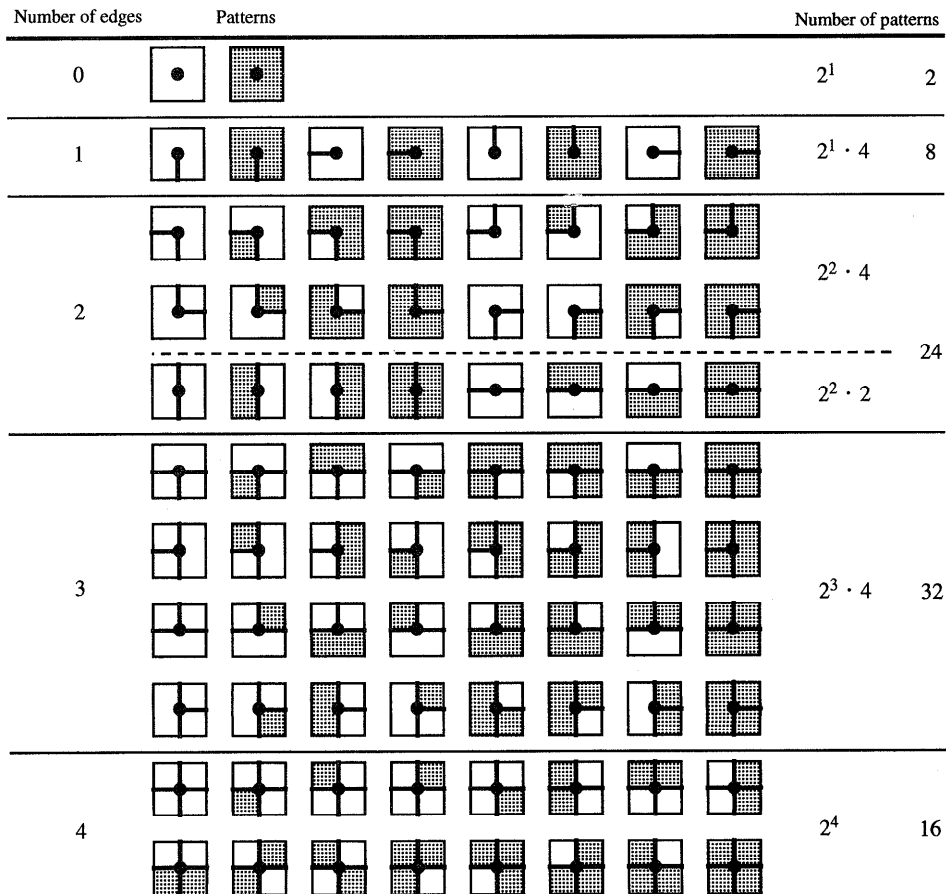


図9 頂点パターンの数え上げ

Fig. 9 Listing up of the vertex patterns.

### 参考文献

- 河合善之, 岡田 稔: CSG 表現からパッチ表現への変換に関する研究, 信学総大, D-12-173 (Mar. 1997).
- 河合善之, 岡田 稔: 多面体表現のための効率的なパッチ生成の一手法, 情報処理学会研究会資料, グラフィックスと CAD, 97-CG-86-3, pp.11-16 (Aug. 1997).
- 金 環垠, 岡田 稔, 横井茂樹, 鳥脇純一郎: 半空間集合の CSG 表現形状データに対する光線追跡アルゴリズムの検討, 信学技報, PRU88-129, pp.33-40 (Feb. 1989).
- 岡田 稔, 金 環垠, 横井茂樹, 鳥脇純一郎: 半空間プリミティブと 3 値論理代数系に基づく多面体の記述, 情報処理学会論文誌, Vol. 38, No.8, pp.1583-1592 (Aug. 1997).
- 久保寺玄一郎, 岡田 稔, 横井茂樹, 鳥脇純一郎: 半空間プリミティブによる対話的 CSG モデラ, 情報処理学会研究会資料, グラフィックスと CAD, 92-CG-55-2, pp.1-8 (Feb. 1992).
- Heckbert, P.S. and Hanrahan, P.: Beam Trac-

ing Polygonal Objects, *Computer Graphics*, Vol.18, No.3, pp.119-127 (1984).

- 7) Pesce, M.: *VRML Browsing & Building Cyberspace*, New Riders Publishing Tomorrow's Realities (1995).
- 8) Matsuda, S.N. and Roehl, B.: *Special Edition Using VRML*, QUE Corporation (1996).
- 9) Hertel, S. and Mehlhorn, K.: Fast triangulation of plane with respect to simple polygons, *Information and Control*, Vol.64, pp.52-76 (1985).
- 10) O'Rourke, J.: *Computational Geometry in C*, Cambridge University Press (1993).
- 11) Koyama, K., Kawai, Y., Tomizawa, Y. and Okada, M.: A Recursive Region Segmentation Method of Projection Screen for High Speed Ray Tracing, *Proc. VSMM'98 - 4th Int'l Conf. on Virtual Systems and MultiMedia*, Vol.1, pp.339-343 (Nov. 1998).

## 付 録

### A.1 冗長な頂点

互いに線型独立な3平面により、全空間は8個の半無限空間に分割され、多面体の頂点を構成する(図3)。そのうち1平面に注目すると、その平面上では、頂点を中心として最大で4本の稜線が存在し、最大で4つの領域に分割される。元々の三次元物体で考えると、この領域は、物体の境界であるときに物体を構成する多角形の一部である。そこで、先の4領域を{境界}:グレー, {外部∪内部}:白として場合分けしたものが、図9である。ただし、これは最大で4方向の稜線で構成される頂点パターンをすべて数え上げたものである。その総パターン数  $N$  は、稜線により構成される領域の1/0による塗り分け、すなわち領域数をビット数とする二進数の数え上げ問題に帰着され、

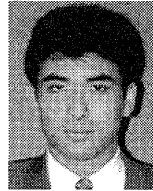
$$N_1 = 2^1 + 2^1 \cdot 4 + 2^2 \cdot 4 + 2^2 \cdot 2 + 2^3 \cdot 4 + 2^4 = 82$$

通りである。このうち、回転対称性、軸対称性を考慮

すると代表頂点パターンは図4に示す  $N_2 = 23$  通り、さらに中央の頂点が存在しなくてもよい、すなわち冗長頂点パターンを除くと、有効頂点パターンは図4(d)に示すような  $N_3 = 7$  通りとなる。

(平成9年12月22日受付)

(平成11年4月1日採録)



河合 善之 (学生会員)

平成9年名古屋大学工学部電気系学科卒業、平成11年同大学院博士前期課程電子情報学専攻修了、同年4月(株)NTTコミュニケーションウェア入社、現在に至る。在学中、コンピュータ・グラフィクス、特に半空間プリミティブに基づく幾何形状モデリング、自然言語による幾何形状モデリングに関する研究に従事。



岡田 稔 (正会員)

昭和59年名古屋工業大学工学部電気工学科卒業、平成元年名古屋大学大学院博士課程後期課程情報工学専攻修了。工学博士。昭和63・平成元年度日本学術振興会特別研究員、平成2年名古屋大学情報処理教育センター助手、平成5年同助教授、平成10年同大学院工学研究科情報工学専攻助教授を経て、平成11年中部大学工学部工業物理学科教授、現在に至る。平成7年より1年間、米国ミシガン大学客員准教授。画像の認識と理解、計算幾何学、画像合成、情報ネットワークの構成理論に関する研究に従事。電子情報通信学会、画像電子学会、IEEE各会員。平成4年度情報処理学会研究賞、平成6年度市村賞学術貢献賞、平成9年度電気通信普及財団賞各受賞。著書「Cによるプログラミング演習」(近代科学社)、「情報科学基礎論」(朝倉書店)、「インターネット時代のコンピュータ活用法」(コロナ社)等。