

## プリミティブの直和及び差演算の CSG を出力する 三面図解釈の一手法

北嶋 克寛

東京農工大学工学部電子情報工学科

田坂 守康

オムロン (株)

三面図解釈の研究はすでに数多く報告されているが、最終的に出力されるソリッドモデルはB-reps表現に限定される方法がほとんどを占める。しかし、CADの入力手段としての応用を考えると、寸法の修正が容易でかつ突起や穴などの幾何学的な特徴を陽に保持するCSG表現が必要になることが多い。CSGからB-repsへの変換は容易であるが、その逆は困難である。このような認識の下に、本研究では、6種類のプリミティブの直和及び差演算で表現される形状をもつ部品の三面図を対象とし、それらの正しいCSG treeを生成することを保証する一連のアルゴリズムの開発を行った。その概要は、各面図に現れる稜線ループに着目し、意味のあるプリミティブ分割を一定の規則の下にすべて解として抽出し、それらの包含関係を調べて体積素の正負を判定し、CSG treeを生成するものである。実験の結果、本アルゴリズムが正しく動作することが確かめられた。

## A Method to Reconstruct a CSG Solid Model from a Set of Orthographic Three Views

Katsuhiko KITAJIMA

Faculty of Technology, Tokyo University of Agriculture and Technology  
2-24-16 Naka-machi, Kogakei-shi, Tokyo, 184 Japan

Moriyasu TASAKA

OMRON Co.

134 Kobe-cho, Hodogaya-ku, Yokohama-shi, Kanagawa, 240 Japan

This paper presents a method to reconstruct a CSG solid model, which is composed of a binary tree of direct sum and difference operations, from a set of orthographic three views of a mechanical drawing. The main stages of the method are, (1) to extract all (under certain constraints) combinations of the loops of 2D primitives for each view, (2) to generate all possible combinations of 3D primitives and (3) to finally generate a CSG tree of direct sum and difference operations. The algorithm is described clearly in detail and some examples are finally shown.

## 1 まえがき

図面理解による3次元形状モデリング法は、すでに多くの報告がなされている[1]。しかし、それらの多くは、出力される形状モデルがB-repsであるという点に特徴がある[2-6]。3次元の形状表現には、よく知られているように、B-repsと並んでよく使われるCSGがあるが、それらの論文において、CSGを生成しない理由について触れているものはほとんど存在しない。この理由を推測すれば、CSGに比べB-repsの方がデータ表現の構成の仕方が直接的で任意性がなく扱いやすいからであると思われる。

ところで、このような研究によって実現されるシステムの実用的な用途を考えると、現状では3次元CADの入力ツールとしての応用への可能性が最も高い。現在の3次元CADにおいては、データ表現として、プリミティブからの形状生成の履歴がわかりかつ穴や突起などの幾何学的な特徴が陽に保存されるCSGの方がB-repsよりも重要な役割を果たす場合も多い。そこで、ほとんどの実用システムでは、B-repsとCSGを併用している。

この分野では、一般にCSGからB-repsへの変換は容易であるがその逆は困難である、という事実がよく知られている。したがって、B-repsだけを生成する認識方法は、その将来の応用用途までを考慮すると問題があるというのが著者らの考えである。

海外では、B-reps[7-9]だけでなく、Surfaceモデル[10]やCSG[11-13]を出力する研究も存在する。プリミティブを対話的に明示する方法、平行掃引によってsubvolumeを求めていく方法などがある。

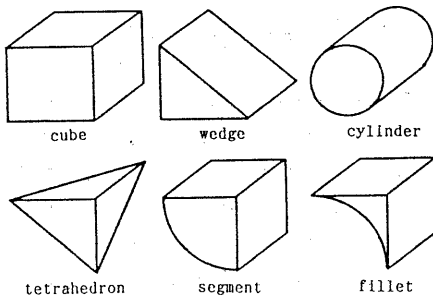


図1 6種類のプリミティブ形状

本研究では、平行掃引を用いずに、三面図上のループの組合せに着目し、機械部品形状によく現れる6種類のプリミティブの組を系統的に抽出しこれらの直和及び差演算のCSG表現を出力する三面図解釈のアルゴリズムの開発を行った。

## 2 前提条件

解釈アルゴリズムの曖昧性を排除するため、本研究では、対象とする三面図及び3次元形状モデルに関して、次のような前提条件を設けている。

- (1) 入力三面図に誤りはないものとする。すなわち、検図は、本研究の範囲外とする。
- (2) 入力データ上では、破線(かくれ線)は実線に置き換えられ、2次元形状はすべて実線で表現される。ただし、複数の解が得られた後に、それから三面図を逆生成しもとの三面図と比較する際は、破線の情報も用いられる。
- (3) BUILD[14]の6種類のプリミティブ(図1)を採用し、これらの直和と差演算のCSGを出力する。すなわち、これらで表現しうる範囲に形状を限定する。
- (4) 上の各プリミティブは、三面図上では図2に示す2次元基本形状のいずれかとして現れる。

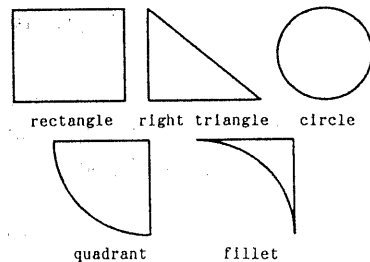


図2 5種類の2次元基本形状

## 3 入力及び前処理

### 3.1 入力データの形式

正面図、平面図及び右側面図の座標系を、それぞれ、 $xz$ 、 $xy$ 及び $yz$ に設定する。図面要素は、頂点及び交点を点(node)とし、直線又は四分円弧からなる線分要素を枝(branch)とするグラフとし

て扱う。直線要素には、両端点の座標値が与えられ、円弧要素には、両端点の座標値、円の中心座標値、半径及び弧の存在する象限の各情報が与えられる。円は4つの四分円弧から構成されるループとなる。

なお、実線と破線の区別はするが、この情報は「7 解候補の正誤判定」の処理まで用いられない。

### 3.2 平面と円柱面及び円柱面同士の境界に対応する枝の付加

隣接する2つの面の境界が不連続に接する場合は、その境界部分は三面図上に枝として現われる。しかし、平面と円柱面又は円柱面同士のようにそれらが連続して接する場合は、その境界部分は三面図上には現われない。したがって、このままでは後の処理においてプリミティブを生成するために必要なループを抽出することができない。そのために事前に次の処理を施す。

- (1) 直線と四分円の枝又は四分円の枝同士が滑らかに接続する点を検出する。
- (2) (1)で検出した点から接線に垂直な方向に線を引き、他の面図中のすべての枝との交点上に点を作成しその間に枝を補う(図3)。

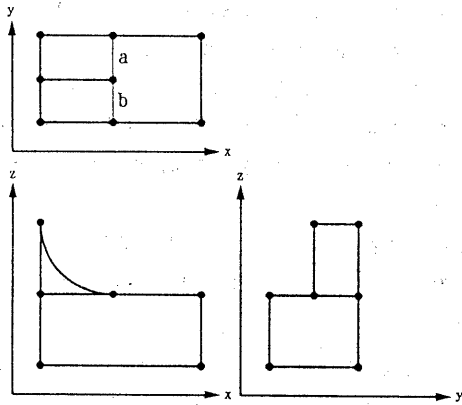


図3 円柱面が他の面になめらかに接続する部分において枝を補う必要のある面図例

### 4 基本形状ループの組合せの抽出

本アルゴリズムの中核は、各面図ごとにループ探索を行いすべてのループを求め、中間的に各種

処理を経て面図ごとにループの組合せを求め、最終的に各面図中のループの組合せを一つずつ選択し三面図間でそれらの対応を取りプリミティブの組を生成することにある。これらの手順の概要を以下に示す。

- (1) プリミティブを構成する三面図の各面図の2次元図形のすべてのループを求める。これは、原始ループ行列をもとにしてすべてのサイクルを代数的に求め、その中から非連結なサイクルと度数が4以上の点をもつサイクルを除去することによって得られる。それらを基本形状ループと非基本形状ループに分ける。前者については、(3)に行く。後者については、はじめに、プリミティブの生成に関係しない不要な非基本形状ループを除去する。
- (2) 残った非基本形状ループに対して枝を補い、複数の基本形状ループに分割する。この分割パターンは一般に複数存在する。
- (3) (得られた複数の分割パターンに対し) 各面図間で基本形状ループの対応を取り、対応の取れる場合にはその基本形状ループの3つ組に対するプリミティブを生成し、対応の取れない場合にはその基本形状ループ(を含む分割パターン)をあらかじめ除去する。これにより、1つの面図における可能なループの組合せの候補の数が大幅に絞られる。
- (4) 次に、ループ分割前から存在する基本形状ループと非基本形状ループを用いてすべての枝を使用するループの組合せを求める。ループの組合せの中に非基本形状ループを含むならば、それをすでに分割しておいた基本形状ループの分割パターンに置き換え、基本形状ループだけからなるループの組合せに変換する。これらの過程においては組合せ処理が基本となるため、効率的な処理を行うためにいくつかの工夫が必要となる。このことを考慮して、不要な非基本形状ループの除去、非基本形状の分割、対応の取れない基本形状ループ(を含む非基本形状の分割パターン)の除去及びループの組合せ処理のそれぞれにおいて、次のような規則を用意し、処

理の効率化を図る。

- ・ 不要な非基本形状ループの除去に関する規則：  
複数の基本形状ループの和に等しくなる非基本形状ループを除去する。

除去してはいけない例外的な場合があるが、ここでは割愛する。

- ・ 非基本形状ループの分割に関する一般的な規則：  
新しく得られる基本形状ループの数が最小となるループの組合せを求める。

- ・ 非基本形状ループの分割に関する特殊な規則：

(1) 斜め線及び四分円弧の枝を含まない非基本形状ループについては、次の規則に従って分割する。

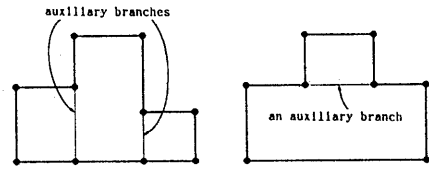
- (i) 内側に枝を補う場合は、内角が  $270^\circ$  の頂点を一方の端点とする枝を補うという規則によって複数の長方形に分割する。この規則に従えば、補う枝の数は、1つの  $270^\circ$  の頂点につき1つの枝となる (図4 (a))。一般に、 $270^\circ$  の頂点の数を  $m$  とすると、新しく得られる長方形のループの数は  $m+1$  となる。ただし、補う枝の端点が2つとも  $270^\circ$  の頂点となるような特殊な幾何学的条件下では、新しく得られる長方形のループの数は  $m+1$  よりも小さくなる (図4 (b))。

- (ii) 外側に補う場合は、ループの外接長方形となる枝を補う。

(2) 斜め線を含む基本形状ループは直角三角形だけである。したがって、非基本形状ループに斜め線の枝が含まれる場合は、斜め線の枝を利用して直角三角形を生成するように枝を補う。この場合も、ループの内側又は外側に補う場合の2通りがある (図5)。

(3) 四分円弧を含む非基本形状ループも、斜め線を含む場合と同様である (図6)。この場合、新しく得られる基本形状ループは四分円又はフィレットとなる。

直角三角形、四分円及びフィレットをすべて生成した後の非基本形状ループが斜め線及び四分円弧を含まない非基本形状ループとなる場合は、(1) のケースとなるので、さらにこれを長方形のループに分割する。



(a) 一般的な場合

(b) 特殊な場合

図4 座標系に平行な直線だけからなる非基本形状ループの分割

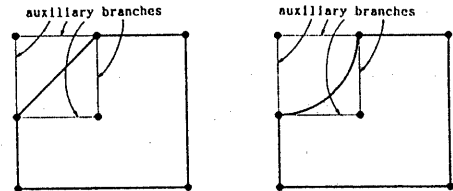


図5 斜め線を含む非基本形状ループの分割

図6 四分円弧を含む非基本形状ループの分割

- ・ 対応する基本形状ループの3つ組の検出とプリミティブの生成及び対応しない不要な基本形状ループの除去に関する規則：  
ある面図中に対応の取れない基本形状ループがある場合は、それ又はそれを含む非基本形状ループの分割パターンをその面図から除去する。

なお、除去された基本形状ループ (の組) に固有の枝がある場合には、この枝も除去する。面図が正しく書かれている限り、除去される枝は付加された枝に限られる。これにより、3.2 の処理で付加された枝のうち、本来の境界を表さない枝が除去される。

- ・ ループの組合せの取り方に関する規則：

複数のプリミティブで構成される立体において、各プリミティブに固有な枝が一面図に少なくとも1つ以上現れていれば、その面図の独立ループ (以下) となるループの組合せを選択することによりすべてのプリミティブを生成することができる。このような一般の場合については、後の(1)、(2)の規則に基づいて、各面図ごとに独立ループ (以下) となるループの組合せを生成する。

ただし、立体を構成する複数のプリミティブのうち異なるループに対応するいくつかのプリミテ

タイプの枝が重なって1つの面図上に現れ、あるプリミティブに固有な枝が面図上に1つも現れなくなる場合がある。この場合には、独立ループとなるループの組合せではすべてのプリミティブを生成することは不可能になる。このような例外的な場合については、処理効率の点から一般の処理の流れの中には含めずに、解が得られなかったときの例外処理とする。これについては後に述べる。

ループの組合せ処理は、残った基本形状ループと非基本形状ループのままで行う。このことは、残った非基本形状ループの分割パターンに含まれる基本形状ループは他の面図と対応が取れることが前の処理によって保証されていることから可能になる。これにより、ループの組合せは、基本形状単位で行うのに比べきわめて少なくなる。

(1) はじめに、対象とする残ったループに対して、次のような場合分けに基づいて処理を行う。

- (i) 残ったループ数  $n$  が独立ループ数  $k$  以上である場合には、独立ループとなる組合せをすべて求める。ただし、含まれる非基本形状ループの数が最も少ない組合せを選択する(図7)。
- (ii)  $n$  が  $k$  よりも小さくなる場合には、残ったループのうち独立となるようなループの組合せを求める。図が正しく書かれているかぎり、このような組合せは必ず残りのすべての枝を用いる組合せとなる(図8)

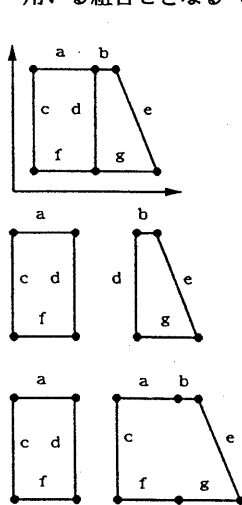


図7 最少の数の非基本形状ループを含む組合せのとり方の例

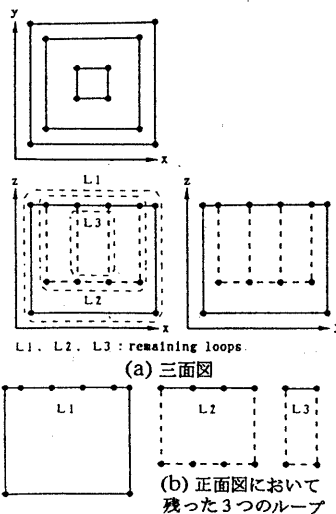


図8  $n$  が  $k$  より小さくなる場合の組合せのとり方の例

(2) (1)で生成された組合せからいくつかのループを除去してもその組合せの中の残りのループがすべての枝を用いているならば、それらのループも組合せに含める。すなわち、すべての枝を用いる  $m$  ((i)の場合は  $m=k$ , (ii)の場合は  $m < k$ ) 未満の数のループの組合せも求める。このような組合せは、図9のように、3つ以上の枝が同一直線上にある場合のように、特殊な幾何学的条件が成り立つ形状において生じる。

(1), (2) で生成したループの組合せの中に非基本形状ループが含まれる場合には、その非基本形状ループを基本形状ループの分割パターンに置き換えてから、次章の処理に進む。次は、独立ループの組合せだけではすべてのプリミティブを生成できない場合に対する例外処理に関する規則である。

(3) 2つの面図A, Bに共通な座標成分の最小値と最大値をもつ面図Aの最大のループ  $L_{A1}$ に着目する。その最小値と最大値にそれぞれ一致する最小値と最大値をもつ他のループ  $L_{A2}$ ,  $L_{A3}$ がその面図中に存在すれば、面図Bのループの組合せに、ループ  $L_{A2}$ ,  $L_{A3}$ に対応するループを含むループの組に従属しかつループ  $L_{A1}$ に対応するループを探し、該当するループがあればそれを付加する。(図10に例を示す。側面図に元々存在しなかったループ  $L_{s4}$ が付加される。)

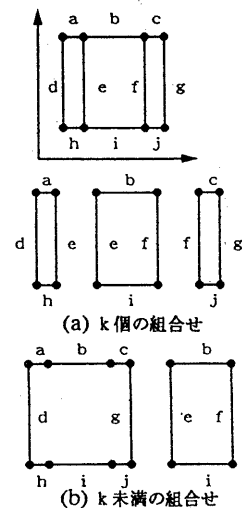


図9  $k$  未満の数のループからなる組合せがとれる例 ( $n=k$ )

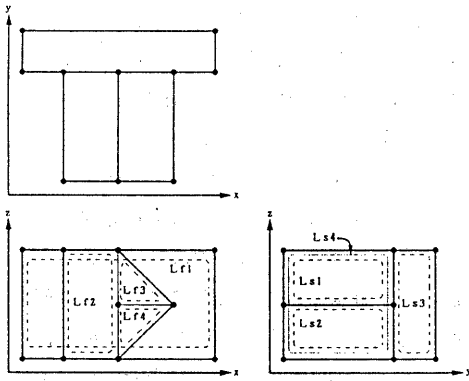


図10 独立ループ数 $k$ よりも多い数のループからなる組合せのとり方が必要になる例

### 5 解候補となるプリミティブの組の生成

次に、各面図に対して得られている基本形状ループの組合せに対し、4章のプリミティブの生成処理で得られているループの対応情報を用いて、3つの面図の各基本形状ループの組合せの中のすべての基本形状ループの対応が取れる各ループの組合せの組すなわち解候補となるプリミティブの組を求める。その手順を以下に示す。

- (1) 各面図ごとに、複数の基本形状ループの組合せの中から、それぞれ1つの組合せを選ぶ。
- (2) 正面図のループの組合せから1つのループを選ぶ。
- (3) ループの対応情報を用いてそのループと他の2つの面図（側面図及び平面図）のループの組合せの中のループとの対応関係を調べる。対応が取れているループがあれば、そのループの組に対応するプリミティブを呼び出しプリミティブの組に加え、(4)に行く。なければ、(1)に戻り、他の面図の別のループの組合せで(1)~(3)を繰り返す。
- (4) さらに、その正面図のループと他の2つの面図のループの組合せの中の(3)で選んだ以外の残りのループとの対応関係を調べ、対応が取れているループがあれば、それに対応するプリミティブを呼び出しプリミティブの組に加える。
- (5) (4)の手順を繰り返し、対応するループがなくなれば、正面図の次のループを選び、(3)へ戻る。

(6) (1)~(5)の手順を繰り返し、正面図に残りのループがなくなれば、解候補となる1つのプリミティブの組の生成を終了する。

(7) 残りの正面図のループの組合せの中から、次のループの組合せを選び、(2)~(6)の手順を繰り返す。解候補となる異なるプリミティブの組が生成されなくなるまで、この手順を繰り返す。

なお、前章で述べたように、独立ループの組合せでは正しいプリミティブの組が生成されない場合には、4章の最後の例外処理に戻った上、再び本章の処理を行う。

### 6 プリミティブの符号の判定及びCSG treeの生成

すべてのプリミティブの生成が終了した後、次に各プリミティブが正か負であるかの判定を行う処理が続く。「2前提条件」から、任意の2つのプリミティブの間の包含関係は、一方のプリミティブが他方を完全に含むか又は全く含まないかのいずれかである。したがって、あるプリミティブが偶/奇数個のプリミティブに含まれていれば、そのプリミティブの符号は正/負となる。なお、負の符号をもつプリミティブが右側に来る直和演算が差演算となる。

前提条件を満たさないプリミティブが存在するときは、プリミティブの符号を判定することができないので、CSGを生成しない。

プリミティブの符号を判定する処理と並行して、プリミティブの結合関係を求めて、CSGtreeを生成する処理を行う。

各プリミティブは、それ自身を含むプリミティブの個数によって、CSGtreeのどの階層に位置するかが決定される。個数が多いほど、上位の階層に位置づけられる。詳しい手順を以下に示す。

- (1) 各プリミティブが何個の他のプリミティブに含まれるかを求める。
- (2) 他のプリミティブに含まれないプリミティブ同士の接合関係を求め、接触しているプリミティブ同士に直和演算の2進木を生成する。

(3) (2)と同様にして、1つのプリミティブに含まれるプリミティブ同士に直和演算の2進木を生成し、(2)で生成された形状との間に差演算の2進木を生成する。

(4) 2つのプリミティブに含まれるプリミティブ同士に直和演算の2進木を生成し、(2)、(3)で演算を施してある形状との間に直和演算の2進木を生成する。

このように、 $n$ 個のプリミティブに含まれるプリミティブ同士に対し直和演算を施し、 $n-1$ 個以下のプリミティブに含まれ、すでに演算を施してあるプリミティブとの間で、 $n$ が偶数ならば直和演算を、奇数ならば差演算の2進木を生成する(図11(b)参照)。すべてのプリミティブを使用したとき、CSG treeの生成を終了する。

(5) 最後に、次の手順に基づいて統合可能な複数のプリミティブを統合する。

(i) CSGtree上で同一の演算記号をもつ任意の2つのプリミティブに対し、それらに直和演算を施して得られる形状がプリミティブ形状になるかどうかを調べる。

(ii) (i)を満たすプリミティブの組が複数ある場合には、2つのうち上位のプリミティブを比較し最も上位のプリミティブをもつ組を見つける。なお、1つの組だけの場合には、□以下ではその組だけを対象とする。そのような組が1つも無い場合には、処理を終了する。

(iii) (ii)で得た組の中で下位のプリミティブに着目する。そのプリミティブとCSGtree上で1つ上位のプリミティブが共通部分をもたないならば、両者の階層を入れ換える。

(iv) (iii)の処理を繰り返し、下位のプリミティブが上位のプリミティブの1つ下の階層に到達したならば、2つのプリミティブを統合し1つのプリミティブとする。到達できないならば、途中まで移動した下位のプリミティブを元の階層に戻す。

(v) 次に、2番目に上位にあるプリミティブの組に対し、(iii)~(iv)の処理を繰り返す。すべてのプリミティブの組に対して(iii)~(iv)の処理

を行ったならば、再び(i)から繰り返す(図11(c)参照)。

## 7 解候補の正誤判定

生成されたCSGtreeは、そのまま解となるものがほとんどであるが、次のような場合に虚のプリミティブを生成する可能性がある。

(1) 得られたすべてのプリミティブのうち、2つ以上のプリミティブを統合することにより得られるプリミティブが解候補の中の他のプリミティブと等しくなる場合、虚のプリミティブを生成する可能性がある。

図12に例を示す。このような場合には、得られたすべてのCSGtreeから三面図を逆生成し、入力三面図とのマッチングを行い正誤判定を行う。

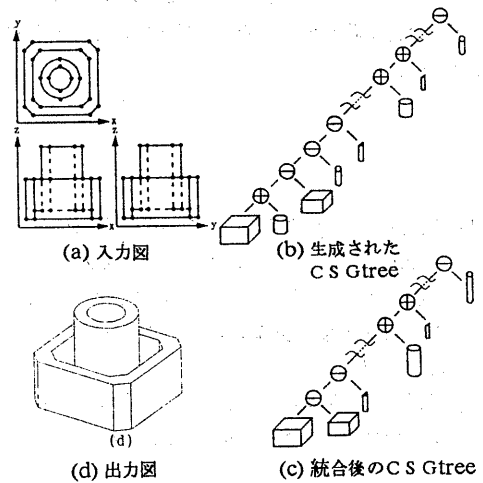


図11 適用例

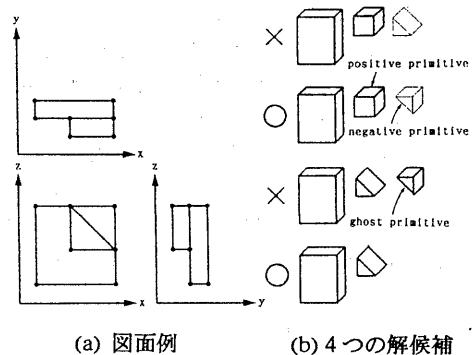


図12 虚のプリミティブを生成する図面例

## 8 むすび

本論文では、三面図を入力情報とし、稜線ループに着目してプリミティブを生成し最終的に直和及び差演算のCSG treeを生成する一連の手法について報告した。

本手法は、機械部品においてよく用いられる6種類のプリミティブとその直和及び差演算によるCSG treeの生成に限定しているが、その条件下で誤りなく解が得られることを保証している。前提条件の下でも適用できる部品形状の範囲は広い。

なお、計算量についての考察などは、文献[18]で行っている。また、著者らは三面図解釈だけでなく、スキャナーから読み込んだイメージデータを出発点とするオフライン機械図面認識の研究も行っており[19-21]、今後も目的と応用を明確にした一貫した図面理解の研究を展開することを目標としている。

## 文 献

- [1] 西原清一: "図面理解による三次元形状モデリング", Computer Today, 7, 56 (1993), pp.19-29.
- [2] 出沢正徳: "三面図からの立体形成のためのシステム", 日本機械学会論文集, 38, 310 (1972), pp. 1267-1276.
- [3] 西原清一, 渡辺恒文, 池田克夫: "面間の局所的拘束関係を用いた三面図解釈", 情処学論, 28, 5 (1987), pp.534-537.
- [4] 佐々木康仁, 伊藤潔, 鈴木誠道: "三面図からの物体自動合成のための線形擬似ブール代数解法", 情処学論, 28, 12 (1987), pp.1288-1297.
- [5] 佐々木康仁, 伊藤潔, 鈴木誠道: "非線形擬似ブール代数解法による三面図からの物体の自動合成", 情処学論, 30, 6 (1989), pp.699-708.
- [6] 千田豊満: "三面図からもとの立体の自動復元", 情報処理学会論文集, 31, 9 (1990), pp.1312-1320.
- [7] Wesley M.A.and Markowsky G.: "Fleshing out projections", IBM J. Res. Dev., 25, 6 (1981), pp.934-954.
- [8] Preiss K.: "Algorithms for automatic conversion of a 3-view drawing of a plane-faced part to the 3D representation", Computers in Industry, 2 (1981), pp.133-139.
- [9] Sakurai H. and Gossard D.C.: "Solid model input through orthographic views", Computer Graphics, 4 (1983), pp.243-252.
- [10] Richards T.C.and Onwubolu G.C.: "Automatic interpretation of engineering drawings for 3D surface representation in CAD", Computer-Aided Design, 18, 3 (1986), pp.156-160.
- [11] Bin H.: "Inputting constructive solid geometry representations directly from 2D orthographic engineering drawings", Computer-Aided Design, 18 (1986), 3, pp147-155.
- [12] Aldefeld B.: "On automatic recognition of 3D structures from 2D representations", Computer-Aided Design, 15 (1983), 2, pp.59-64.
- [13] Chen Z.and Perng D.B.: "Automatic reconstruction of 3D solid objects from 2D orthographic views", Pattern Recognition, 21, 5 (1988), pp439-449.
- [14] Braid I.C.: "Designing with volumes", Cantab Press, (1973).
- [15] 北嶋克寛, 岩谷隆雄: "プリミティブ分割に基づく三面図解釈", 精密工学会講演論文集, H36 (1988秋), pp.555-556.
- [16] 北嶋克寛, 田坂守康: "プリミティブ分割に基づく三面図解釈 (第2報) -対象立体の拡大-", 精密工学会講演論文集, H55 (1991春), pp.755-756.
- [17] 北嶋克寛, 山口満徳: "面の境界稜線ループに基づく境界表現に適する立体の集合演算アルゴリズム", 信学論(D-2), J74-D-2, 8 (1991), pp.1070-1079.
- [18] 北嶋克寛, 田坂守康: "プリミティブの直和及び差演算のCSGを出力する三面図解釈の一手法", 電子情報通信学会論文誌D2, J75-D-2, 9 (1992), pp. 1526-1538.
- [19] 北嶋克寛, 黒岩邦博, 志田吉忠: "機械製図図面の認識 (第1報) -線図形要素分離のためのボトムアップ処理-", 精密工学会誌, 56, 11 (1990), pp.2057-2062.
- [20] 北嶋克寛, 志田吉忠, 黒岩邦博: "機械製図図面の認識 (第2報) -幾何学的及び位相的特徴に基づく線要素の分離", 精密工学会誌, 58, 1 (1992), pp.169-176.
- [21] 北嶋克寛, 諏訪光信, 岩谷隆雄: "機械製図図面の認識 (第3報) -図面構造モデルに基づくトップダウン処理-", 精密工学会誌, 58, 10 (1992), pp.1721-1727.