

(昭和 63 年 11 月 日本造船学会秋季講演会において講演)

# 統合化 CAD/CAM 構築のための 船体構造表現の基礎的研究

正員 小山 健 夫\* 正員 秋 藤 俊 介\*\*

A Fundamental study on representation of ship's hull structure for  
integrated CAD/CAM in shipbuilding

by Takeo Koyama, *Member* Shunsuke Akifuji, *Member*

## Summary

Integrated CAD/CAM is required in many industrial fields. Building integrated CAD/CAM for ship-building, we have to consider of characteristic problems such as how to input many detailed information of parts and how to measure against frequent design changes. In this paper we propose a description method of hull structures by relation and explain geometrical reasoning to fix figures from this description.

We also propose a parts decomposition method applying the simplifying and integrating rules. Our experimental program is able to deal with objects consist of planes only, but improved rules are needed to put this program into practical use with curved surfaces.

## 1 緒 言

現在までに、さまざまな分野において設計・生産の効率化のためにコンピュータが使用され、多くの CAD/CAM システムが作られている。しかし、それらのシステムは、設計から生産までの流れの中でのあるステップで個別に使用されている場合が多く、設計上流から実際の加工情報を生成するような統合的なシステムになっていないのが現状である。今後、複雑化する製品に対する設計・生産コストの上昇、納期の短縮等の諸問題を解決するためには、CAD と CAM の統合化、つまり統合化 CAD/CAM システムが必要である。この統合化 CAD/CAM システムを構築するために、木村らは、プロダクトモデルを基礎とする方法を提案している<sup>1)</sup>。

ところで、船舶に関しては、次のような特徴的な問題点があり、統合化 CAD/CAM システムを作り、利用することを困難にしている。

i) 短納期であるため、設計展開が不十分なうちに分担作業が開始されるために設計変更が多い。設計変更によって全体の寸法や形状に矛盾が生じてはいけませんが、これをどのように効率良く解決するか。

ii) 大規模であるために部品数が多く、形状や接続関係のデータを細かく定義することが困難である。

iii) 船体構造は、類似船形の場合でも寸法だけでなく位相が異なっていることが多く、あらかじめパターン化しにくい。

本研究では、以上の問題を解決し、造船に対する統合化 CAD/CAM システムを構築するために、設計・生産において重要な形状の表現法に重点をおいて、次のような方針をとった。

i) パラメタを使用して、面と面の関係を記述する。

ii) 集合演算を用いて形状を定義する。

iii) 設計過程においては、しばしば寸法値が変更されるが、これに対応するためにパラメトリックデザインを実現する。

iv) 部品形状の記述を自動生成するようにする。

i), iii)は、木村らのプロダクトモデルを参考にしたが、面の定義が異なる。また、パラメトリックデザインに関しては、位相情報を入力するものではなく、処理の過程で生成されるものと考え、パラメタの変更によって位相の変化するような場合においても対応できるようにした。集合演算を用いて形状を表現することは、沖野らの立体の表現法 (CSG)<sup>2)</sup>と同様であるが、後で述べるように式を使った独自の表現方法を用いている。

部品の形状や接続関係のデータ定義を自動化するため

\* 東京大学工学部

\*\* 日立製作所システム開発研究所  
(研究当時 東京大学大学院)

に iv) が必要になる。本研究では、集合演算とルールを用いることにより部品形状記述の自動生成を試みる。

なお、本研究のために作った試作システムは、主に Prolog で記述したためルール等の説明の際に Prolog による記述を用いる。

## 2 形状の記述法

はじめに入力する形状記述の方法について説明する。本研究では、パラメタは面の拘束条件を与えるものとして考えている。本研究で扱う面は、無限かつ法線ベクトルを定義でき、面の表裏の区別の可能な面である。形状を定義する際に、まず必要な面（境界となる面と補助的に使用する面）を定義し、この面によって決まる半空間を点の集合として考え、この半空間の集合演算の結果として、3次元の形状を定義する。このような二段階の方法をとったのは、船体構造が主に板材からなっているために、ある二面をとった場合に向かい合う面が等間隔である場合が多く、どちらか一方の面を入力することにより形状定義を容易かつ正確にできると考えたからである。形状を記述するために点列を入力する場合では、陽に位相を扱うことになるために、すでに入力段階で以後変更の困難な情報を形状に加えることになってしまう。

本研究では、形状記述を SRD (Surface Relation Description) と呼ぶ。SRD は、パラメタ宣言部、面関係記述部、集合演算部から成る。これを Fig.1 に示す。本来、集合演算部は面の関係を記述するものではないため、SRD という表現は誤りであるが、便宜上この名称を使用した。

パラメタ宣言部は、使用するパラメタ名と初期値を定義する。面関係記述部は、複数の面関係記述子 SRP (Surface Relation Predicate) を記述してあり、集合演算部では、面によって定義される空間の集合演算による式を記述する。

### 2.1 SRP の定義

現在、次の5種類の SPR を定義している。すべて3

```
beginof(a).
parameter(s1,15).
parameter(t1,1.5).
parameter(h,11).
xref(x).
yref(y).
zref(z).
distance(a,x,1).
distance(b,y,7).
distance(c,z,1).
board(a,d,s1 + 2 ).
board(b,e,t1 / ( 3.0 + s1 ) ).
board(c,f,h * 2 ).
bool(a*b*c*d*e*f).
endof(a).
end.
```

Fig.1 Surface Relation Description

次元直交座標系で定義する。

- ①  $xref(x)$ ,  $yref(y)$ ,  $zref(z)$
- ②  $distance(x,y,p)$
- ③  $board(x,y,p)$
- ④  $angle(x,y,a,b,p)$
- ⑤  $cylinder(r,a,b,p)$

ただし  $x,y,z,a,b$ : 平面の名前,  $r$ : 円筒面の名前,  
 $p$ : パラメタを含む算術式

- ①  $xref(x)$ ,  $yref(y)$ ,  $zref(z)$  (Fig.2 参照)

形状を記述するための基準面を定めるために使う。 $xref(x)$  は、原点を通り  $x$  軸に垂直、法線ベクトルが、 $x$  軸の負の向きと一致する平面を  $x$  とする、という意味である。 $yref$ ,  $zref$  はそれぞれ  $y$  軸,  $z$  軸に関して同様。

- ②  $distance(x,y,p)$

平面  $x,y$  は、互いに平行かつ同じ向き、二面間の距離が  $p$  である。平面  $x,y$  のどちらか一方がすでに固定されており、もう一方がこの記述により固定される。

パラメタ  $p$  の正負によって Fig.3 のような位置関係になる。よって

$$distance(x,y,p) = distance(y,x,-p)$$

- ③  $board(x,y,p)$

平面  $x,y$  は互いに平行で逆向き、二面間の距離が  $p$

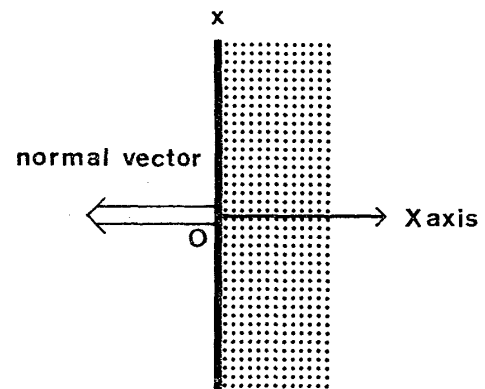


Fig.2 Definition of reference surface

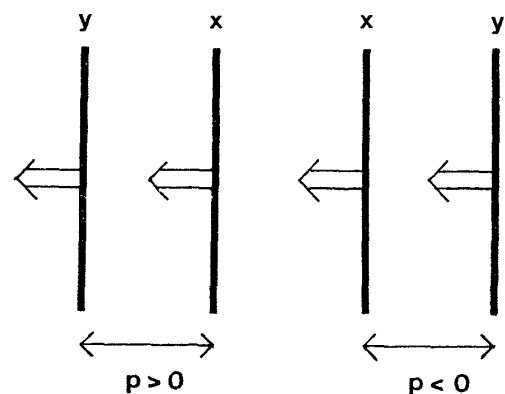


Fig.3 Definition of distance by surface relation predicate

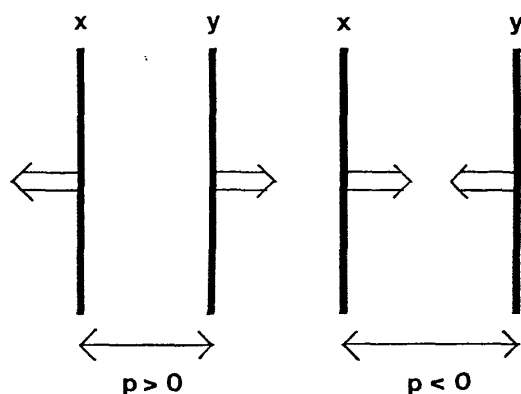


Fig. 4 Definition of board by surface relation predicate

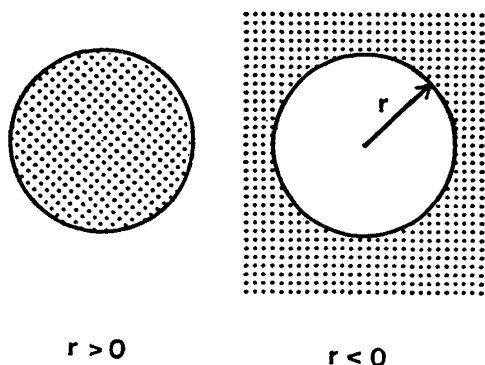


Fig. 5 Definition of cylinder by surface relation definition

である。面  $x, y$  のどちらか一方がすでに固定されており、もう一方がこの記述により固定される。

パラメタ  $p$  の正負によって Fig. 4 のような位置関係になる。よって

$$\text{board}(x, y, p) = \text{board}(y, x, p)$$

board は板の向かい合う面を記述することからこの名前をつけた。

#### ④ angle ( $x, y, a, b, p$ )

平面  $a, b$  は補助面で、平面  $a, b$  の交線を軸として平面  $a, b$  の法線ベクトルの外積に対して角度  $p$  (degree) だけ平面  $y$  を回転した平面が平面  $x$  である。

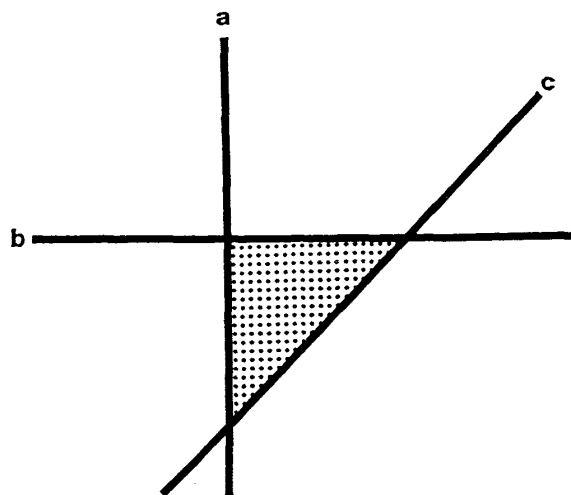
#### ⑤ cylinder ( $r, a, b, p$ )

平面  $a, b$  は補助面で、平面  $a, b$  の交線を軸として半径  $p$  の円筒面が面  $r$  である。半径  $p$  の正負によって面  $r$  を境界として定義される半空間は Fig. 5 のようになる。

SRP は面を定義だけではなく、ある関係を満たす面を探すために使用することも考えられるが、試作システムでは部分的にこれを実現している。

### 2.2 集合演算

SRP を用いて固定される面を境界として、法線ベクトルと逆向きの半空間を、その面によって固定される空間とする。するとこの境界面を示すことによってある空間を定義できるので、集合演算部では面の名前を使って



$$(a + b) * \text{neg}(c)$$

Fig. 6 Definition of space by set calculation

形状を定義する。

集合演算として、和 (union), 積 (inter section), 差 (difference) が考えられる。これを次のように定義する。

union ( $a, b$ ): 境界面  $a, b$  によってそれぞれの定義される空間の和

intersection ( $a, b$ ): 境界面  $a, b$  によってそれぞれ定義される空間の積

difference ( $a, b$ ) = intersection ( $a, \bar{b}$ )

$\bar{b}$  は境界面  $b$  によって定義される空間でない空間

試作システムでは、和、積、差を次のような記法で表現している。

$$\text{union}(a, b) \iff a + b$$

$$\text{intersection}(a, b) \iff a * b$$

$$\bar{a} \iff \text{neg}(a)$$

$$\text{difference}(a, b) \iff a * \text{neg}(b)$$

演算の優先度を  $\text{neg}() > * > +$  の順にし、通常の算術式のような記法によって形状の定義を行う。例を Fig. 6 に示す。

### 3 試作システム

SRD に記述したパラメタ, SRP, 集合演算によって定義される形状をコンピュータ中で固定する方法と、パラメタを変更した時に自動的に形状を再固定するパラメトリックデザインについて説明する。

形状要素の幾何学的拘束関係から形状要素の幾何学的属性を導く過程を幾何学的推論と呼び、本研究においては、SRD から面の式、交点の座標・交線を幾何学的推論により導いている。

### 3.1 幾何学的推論

幾何学的推論によって SRD から具体的な面、交点、交線を固定していく。試作システムでは、① SRD から面を固定する、②面から交点、交線をこの順序に固定する、の二段階にわけている。二段階にわけた理由は、ある面を固定する時に必要な面は、それを固定する SRP を参照すれば決定できるが、交点や交線を固定する時に必要な面は、他のすべての面と交わるかどうかを確認する必要がある、すべての面の情報が固定されている必要があるからである。また交点を交線より先に求めるのは、計算誤差の影響を少なくするために、交線は交点を結んだものとして固定される。

SRD では、面を固定する順序が陽に示されていないので、この順序を求める必要があるが、この種の問題はプロダクションシステムを用いることにより解決できる。試作システムにおいてもプロダクションシステムを用いているが、試作システムの推論エンジンで特徴的なことは、ルールの優先順位をもたず、推論停止条件として、どのルールと SRP を用いても現状以上に面を固定できない時に推論を停止するという条件を与えていることである。このような停止条件にしておくと 3.2 で述べるように、パラメタ変更の際にパラメタによって直接または間接的に関係する面を消却してからパラメタの値を変化させれば、自動的に不足の面、つまりパラメタによって変更される面が新しく固定されるようにできる。このような停止条件を与える場合に注意しなければならない点は、ルールが無限に面を固定するようなルールになってはいけないということである。次に具体例を使って面固定ルールと交点・交線固定ルールを説明する。

#### i) 面固定ルール

SRP として angle があった場合に、これにマッチするルール fixed-angle を Fig. 7 に示す。fixed-angle では、面 Y, A, B がすでに固定されていて面 X が固定されていない時にパラメタとすでに固定されている面の情報を用いて計算を行い面 X を固定する。

#### ii) 交点・交線固定ルール

互いに平行でない 3 平面の交点を固定するルールの例を Fig. 8、2 平面の交線を固定するルールの例を Fig. 9 に示す。Fig. 8 のルールは、任意の互いに平行でない 3

```
rule(
  fixed_angle,
  [angle(N,X,Y,A,B,Expr),
   fixed(Y,_,_,_),
   fixed(A,_,_,_),
   fixed(B,_,_,_),
   \+ fixed(X,_,_,_)],
  [sub_angle(N,angle,[Y,A,B],X,Expr)]
).
```

Fig. 7 An example of surface fixing rule

```
kouten2:-
  plane(A,[Ax,Ay,Az,Ad]),
  plane(B,[Bx,By,Bz,Bd]),
  A @> B,
  \+ para(A,B),
  plane(C,[Cx,Cy,Cz,Cd]),
  B @> C,
  \+ piont([A,B,C],[_,_,_]),
  \+ para(A,C),
  \+ para(B,C),
  .
  .
  .
  include volume([A,B,C],[X,Y,Z]),
  asserta(point([A,B,C],[X,Y,Z])).
```

Fig. 8 Intersection point fixing rule with 3 surfaces

```
kousen2:-
  point(L1,P1),
  point(L2,P2),
  \+ line(L1,L2,_,_),
  intersection(L1,L2,[S1,S2]),
  plane(S1,_,_),
  plane(S2,_,_),
  union(L1,L2,L3),
  bool(Bool),
  includeb(L3,Bool),
  asserta(line(L1,L2,P1,P2)).
```

Fig. 9 Intersection line fixing rule with 2 surfaces

平面を選び出して、その交点がまだ固定されていない時にその交点の座標を計算し、3 平面の定義する空間中にあればそれを新しい交点として固定するという意味である。平面が互いに平行であるかどうかは、SRP を探索して平行であることが証明されるかどうかで判定しており、数値計算で処理していない。交点が面によって定義される領域内にあるかどうかは、数値計算を、その交点を決定した平面の名前から SRD の集合演算を参照して判定しており、境界上にある場合の微妙な誤差の入ることを防いでいる。

Fig. 9 のルールは、任意の 2 点を選び出し、それぞれの交点を固定するために使用した 3 平面のうち 2 面が一致する場合にその交線を固定するルールで、これも集合演算部を参照する。

### 3.2 パラメトリックデザイン

はじめに面がなにも固定されていない時に SRD から幾何学的推論によって面・交点・交線を固定する場合には、それぞれの面・交点・交線と SRD の間に矛盾する関係は生じない。しかし、あるパラメタを変更すると、これらの間の関係に矛盾を生じる。

パラメトリックデザインを実現するためには、この矛盾を解消することが必要である。試作システムでは、ある面・交点・交線を固定した時に固定するために直接使用した SRP、面の名前、パラメタ名を記録し、この記録をたどることにより矛盾が生じないようにしている。

また、この記録をたどることにより、パラメタ変更に関係のない面・交点・交線は処理する必要がないことがわかるため、実行効率を上げることができる。このように事実に変化を生じた時、ここではパラメタの値が変化した時に自動的に矛盾を解消するシステムを TMS (Truth Maintenance System) という<sup>3),4)</sup>。

試作システムでは、ひとつずつ矛盾を解消するのではなく、パラメタ変更によって直接変更される面とその面が固定されていることを条件として固定された他の面・交点・交線をすべて消却し、その後幾何学的推論を始動するようになっている。すると、3.1 で説明したように推論エンジンは消却された面をうめるように動作し新しく SRD と矛盾することのない面・交点・交線が固定される。

### 3.3 実行例

試作システムの実行例を Fig. 10, Fig. 11 に示す。

Fig. 10 は、5 種の SRP すべてを使用した例で、パラメタを変化させた時にこのように形状が変化する。円筒面の周囲に必要な線があるが、これは交線固定ルールが不完全なために生じた。

Fig. 11 は、2 枚の板が直交している場合を想定している。パラメタ変更によって位相が変化していくが、試作システムではこのような場合にも十分対応できることがわかる。交線として本来存在しない線が描かれているが、これは交点固定ルールが不完全であるからである。

以上のように形状を記述する方法と、それから実際の形状を固定する方法を説明した。次にこれらを利用した部品の自動分割について説明する。

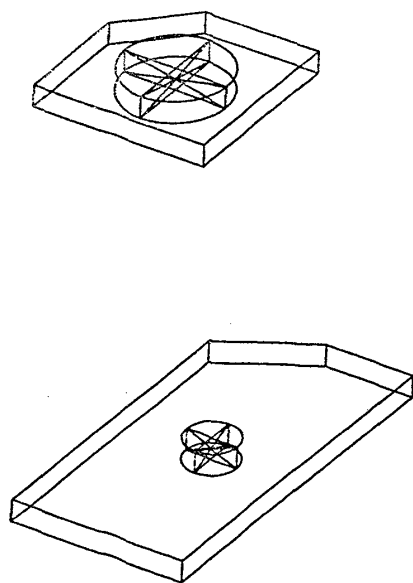


Fig. 10 Examples of parametric representation

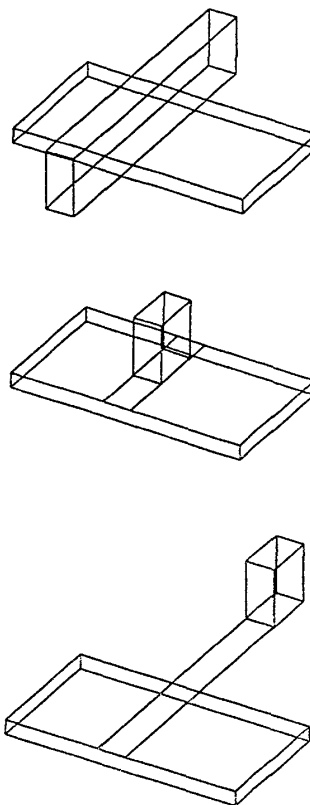


Fig. 11 Parametric representation with phase change

## 4 部品分割

部品とは、一枚の板等の溶接やねじで接続していないひとつのまとまりと定義と、一枚の板を曲げたもの、穴をあけたものはそれぞれひとつの部品であるとする。アングル材のようなものは、一枚の板を曲げて作った場合はひとつの部品であるが、二枚の板を接続した場合には二つの部品である。形状が複雑な場合は、目的に応じて仮想的に部品分割するものとする。

実際の設計・生産において細々とした部品形状や、接続関係、パラメタ間の関係を正確に入力することはコストがかかるため、この自動化が望まれる。部品形状の記述や接続関係の記述を完全自動生成することは難しいが本研究では SRP として distance と board だけを使用して記述した、板と板が交わる場合に限り、つまりすべての面は平面であり、互いに平行または直交する関係にある場合について部品形状の自動生成を試みた。パラメトリックデザインと組み合わせると、パラメタの値によって板が一枚もしくは複数に分割される場合があるが、試作システムではこのような場合にも対応できるようにした。

### 4.1 部品分割の方法

試作システムでは、2 章、3 章で説明した方法で定義された 3 次元立体形状にそれぞれ名前をつけ、その名前

```

make_simple(L1,[]):-
  member(A,L1),
  plane(A,[Ax,Ay,Az,Ad]),
  Bx is -Ax,
  By is -Ay,
  Bz is -Az,
  plane(B,[Bx,By,Bz,Bd]),
  member(B,L1),
  A \== B,
  Ad >= -Bd.
make_simple1([A|R],[A|T]):-
  plane(A,[Ax,Ay,Az,Ad]),
  plane(B,[Ax,Ay,Az,Bd]),
  Ad >= Bd,
  member(B,R),
  delete_all(B,R,R1),
  make_simple1(R1,T).
make_simple1([A|R],T):-
  plane(A,[Ax,Ay,Az,Ad]),
  plane(B,[Ax,Ay,Az,Bd]),
  Ad < Bd,
  member(B,R),
  make_simple1(R,T).

```

Fig. 12 Simplification rule for decomposition

を用いて立体の優先順位を指示する。入力されたすべての立体に対して集合演算の差を用いて優先順位の低い立体と高い立体の差をとる。例えば、立体  $a, b$  がある場合、優先順位を  $a < b$  とすると、集合演算の差を用いて生成される立体  $a', b'$  は

$$a' = a - b = a * \text{neg}(b)$$

$$b' = b$$

となる。得られた結果は集合演算の式であるが、これを展開し、簡略化ルールを用いて空集合になる項、同一項を整理する。簡略化ルールの例を Fig. 12 に示す。

簡略化ルールを用いて整理された結果は複数のプリミティブの和として得られる。プリミティブとは、これ以上分割できない基本立体で、試作システムでは直方体である。

この段階では、いくつかのプリミティブが空間的に連続するかどうか不明なため、実際の部品として認識できない。そこで、ここに一体化ルールを用いて連続するプリミティブをひとつにまとめて部品とするようにする。プリミティブの連続・不連続の判定は、2つのプリミティブを取り出した場合に重複する空間があるかどうかを集合演算の積を使って求め、その空間が空であるか空でないかを調べることで判定している。

試作システムでは、すべての面は平面で、平行または直交する関係にあるだけなので、簡略化ルール、一体化ルールとも数値の比較のみで満足に動くが、本来は幾何学的推論を利用して厳密な判定を行うべきである。

#### 4.2 部品分割の例

3枚の板  $p1, p2, p3$  からなる例を Fig. 13, Fig. 14 に示す。Fig. 13, Fig. 14 では  $p1$  の高さのみが異なる。優先順位は  $p1 > p2, p3 > p2$  である。

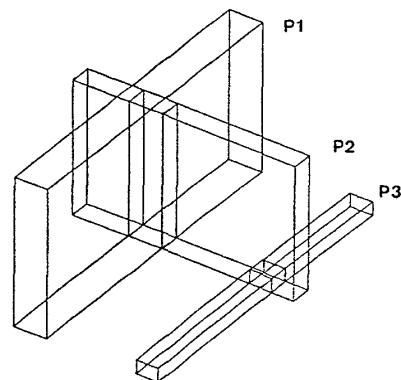


Fig. 13 Example of parts decomposition (1)

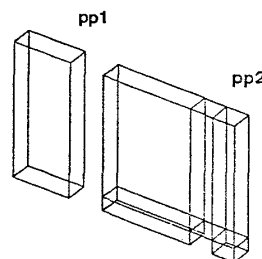


Fig. 14 Example of parts decomposition (2)

$p2$  に注目すると、 $p1$  の高さが増えると2枚に分割する場合と、一枚の板に穴をあける場合の2通りが生じることがわかる。実行例で示されるように試作システムでは、この2つの場合を区別して対応しており、それぞれ生成される形状記述は異なる。現状では部品分割の結果として  $pp1, pp2$  の SRD の集合演算部を出力するだけであり、面の整理やパラメタの合成等を行っていない。

ない。

今回は平行と直交する関係にある平面のみを考えたが、angle を使って平面に角度を与えた場合には、ルールを増加させることによって対応できると考えている。

## 5 考 察

以上、形状記述の方法と幾何学的推論により形状を固定する方法、位相変化に対応できるパラメトリックデザイン、優先順位を与え、ルールを用いて部品分割する方法を説明した。平面だけからなる形状の場合では試作システムが十分対応できることを示したが、円筒面に関しては形状を固定する段階にとどまった。また交点・交線固定ルールは不完全であり今後改良する必要がある。

部品分割に関しては、優先順位を与える方法をとったが、この順位は設計知識をもったエキスパートシステムにより決定されるようにできると考えている。

試作システムの処理時間は、幾何学的推論に基づく形状の固定、部品分割とも非常に長い。処理時間が長い理由は、交点・交線の固定では、パラメタ変更の際、多くの交点・交線が影響を受けるため処理時間が増加するためであり、部品分割では、簡略化ルールを用いるため式を展開すると項数が爆発的に増加するためであり、これを改良するために一種の学習能力をもたせることや、項を一度に展開せず、少しずつ処理していくことが考えられる。

試作システムは平面と円筒面を扱ったが、船体は自由曲面が多く使用されており、これをどうするかが今後の問題である。

船体のように大規模な単品を作る場合には、前もって組立順序や組立作業の細部を決定しておくことが有効であるが、このためにコンピュータを使った組立作業のシミュレーションを行うことが考えられる。組立作業のシミュレーションでは作業中に物と物とが衝突しないことを確認しなければならず、干渉部の検出が大きなポイントになる。本研究で使用した集合演算と簡略化ルールは、この干渉部の厳密な検出に利用できると考えている。

設計・生産の情報の流れを見ると、仕様からパラメタがどのように決定するのか、未だに明らかではないが、これを明らかにすることによって設計知識をもったエキ

スパートシステムを作ることができ、仕様から生産情報を生成する本当の統合化 CAD/CAM システムができると考えている。

## 6 結 言

本研究により、得られた知見は以下の通りである。

(1) 面関係と集合演算を用いた形状記述法を提案した。

(2) この形状記述から形状を固定するために幾何学的推論を使用することにより、位相変化に対応できるパラメトリックデザインを実現できる。

(3) TMS を採用することにより、パラメタと面・交点・交線との間の関係に矛盾のないシステムを作ることができる。

(4) 幾何学的推論を用いているために計算誤差の影響が少なく、将来ルールを増加、改良することにより、適用できる形状の範囲を拡大できる。

(5) 部品分割では簡略化ルール、一体化ルールを用いることが有用である。

このシステムで試みた構造は極めて単純なものに限られており、実際の船体構造とはほど遠い。それにも拘らず相当な処理時間を要する。実用化のためには、不要な幾何学的推論を如何にして排除するかが最大の問題である。現在は定義した面全てを推論の対象としているが、設計対象を特定すれば、個々の面が持つ特性により推論の対象を大幅に限定することが出来ると考える。

## 参 考 文 献

- 1) 鈴木宏正, 木村文彦, 佐田登志夫: プロダクトモデルに基づく幾何学的拘束関係の記述と寸法処理への応用, 精密工学会誌, Vol.52, No.6 (1986).
- 2) 沖野教郎: 自動設計の方法論, 養賢堂 (1982).
- 3) 安藤英俊, 鈴木宏正, 木村文彦: 非単調論理に基づく幾何学的推論, 昭和 61 年度精密工学会秋季大会学術講演会論文集, (1986) pp. 445~446.
- 4) Doyle, J.: Truth Maintenance Systems for Problem Solving, Artificial Intelligence Laboratory, MIT, Tech. Rep. AI-TR-419 (1978).  
(中川幹夫訳, TMS による問題解決, 認知科学の基底, 産業図書 (1986).)