

NC 加工における配置要素の形状特徴を利用した板取システム

金井 秀明^{†1} 若林 伸和^{†2}
本多 中二^{†3} 田中 功二^{†4}

配置問題は、VLSI 素子の配置、プラントレイアウトや GUI のウィンドウ配置など、様々な分野で議論されている。これらの問題では配置要素の組合せが多数で、一般に NP 完全問題となる。本論文では、配置問題のうち NC (Numerical Control: 数値制御) 加工での部品の配置問題 (板取問題) を扱う。これは、一定の材料からできるだけ多くの部品を取り出す問題であり、NC 加工の効率化には欠かすことのできない重要な問題である。板取問題に関する研究は長年にわたって行われ様々な手法が提案され、生産の効率化に寄与してきた。しかし、それらの手法には、(1) 配置要素として矩形状を中心に処理している；(2) 各配置要素ごとに配置ルールを用意する必要があり、新規の要素を配置するには、その最大矩形領域を必要とする；(3) 過去に板取した結果を再利用できない、などの問題点がある。本論文ではこれらの問題に対する解決法の 1 つとして、配置要素の形状特徴を利用した板取手法を提案し、試作したシステムを用いて本手法の有効性について述べる。本手法では、システムが配置要素の形状を認識するために、事例ベース推論によって形状認識を行い、その際、形状認識を補完するために形状近似を行う。認識された要素を、その形状特徴から 5 つのタイプに分類し、各タイプごとに局所的に配置計画を行う。最終的に各タイプの要素を総合して配置計画を行う。

A Nesting System for NC Machining Using Structural Features of Parts

HIDEAKI KANAI,^{†1} NOBUKAZU WAKABAYASHI,^{†2} NAKAJI HONDA^{†3}
and KOJI TANAKA^{†4}

The layout problem has been examined in many different fields such as VLSI, plant and windows of GUI layout design. However, it is known to be NP-complete and therefore it is not possible to provide exact solutions. This paper addresses a packing problem of NC (Numerical Control) machining called the nesting problem. The problem is to allocate as many parts as possible in a certain area. The solution is essential to increase NC machining productivity. There are many applications to propose techniques to the solution and they have contributed to the improvement of productivity. However, they tend to concentrate on the problem of packing rectangular shapes, have to treat a new part as its maximum rectangular area, and can not use the past nesting results easily. In this paper, we propose a solution based on structural features of parts, and show the effectiveness using a nesting system based on our methodology. In our system, we adopt CBR (Case Based Reasoning) including mechanism of approximating structure of parts to recognize their structure. The recognized parts are classified into five types of shape by the structure features, and then for parts of each type, the local positions are decided. Finally, the parts are placed globally on an area.

^{†1} 電気通信大学大学院情報システム学研究科
Graduate School of Information Systems, University of
Electro-Communications

^{†2} 静岡大学工学部システム工学科
Department of Systems Engineering, Faculty of Engi-
neering, Shizuoka University

^{†3} 電気通信大学電気通信学部電子情報学科
Department of Communications & Systems, University
of Electro-Communications

^{†4} 真如苑企画情報管理課
Computer Programming Section, Planning Division,
Shinnyo-en

1. はじめに

ある決められた領域・空間内に、複数の任意形状の物体を適切に配置する問題は、様々な分野で議論されている。たとえば、VLSI 素子の配置、プラントレイアウト¹⁾、機械設備のレイアウト²⁾や、GUI のウィンドウ配置などである。これらの問題は、各問題領域における制約や配置位置の組合せが多数存在するため、NP 完全問題に属する。本研究では、配置問題のうち NC (Numerical Control: 数値制御) 加工での部品の

配置問題(板取問題と呼ばれる)を扱う。これは、一定の材料から、できるだけ多くの部品を取り出す問題であり、NC加工の効率化には欠かすことのできない重要な問題である。

板取問題に関する研究は長年にわたって様々な手法が提案され³⁾、生産の効率化に寄与してきた。しかし、それらの手法には、(1)配置要素として矩形状を中心に処理している；(2)各配置要素ごとに配置ルールを用意する必要がある、新規の要素を配置するには、その最大矩形領域を必要とする；(3)過去に処理した結果を再利用できない、などの問題点がある。

本論文ではこれらの問題に対する解決法の1つとして、配置要素の形状特徴を利用した板取手法を提案し、試作したシステムについて述べる。

以下、2章では板取問題と板取システムの現状について述べる。3章では提案するシステムの概要について述べ、4章、5章でその手法について説明する。6章で試作システムによる実行例を紹介し、システムの有効性を示す。

2. 板取問題について

2.1 板取問題の位置付け

NC工作機械は、現在では生産現場において不可欠な存在である。従来の工作機械が熟練技術者によって人手で操作されるのに比べ、NC工作機械は、制御装置に対してNCデータと呼ばれるプログラム形式の命令を入力することで、ほとんど人手を必要とせずに材料を様々な形状に加工できる。また、最近では、CADによって生成したCADデータからNCデータを自動生成するシステム⁴⁾や材料自動供給システムなどによって、FMS(Flexible Manufacturing System:フレキシブル生産システム)として多くの産業で利用されている。

NC工作機械の1つにタレットパンチプレス(NCT)がある。これは様々な形状の“金型”と呼ばれる工具を用いてパンチ(穴あけ)加工をすることにより、材料である板金を打ち抜き、製品を生産する機械である。NCT用のNCデータ自動生成システムが開発されているが、人間の介在が不可欠な部分がある。その1つに、製品を材料のどこへ配置するのかという問題がある。これは板取問題と呼ばれ、一定の材料から、できるだけ多くの部品を取り出すための部品配置を決める問題で、NC加工の効率化には欠かすことのできない重要な課題である。

2.2 板取システムの現状

板取問題に関する研究は、手法の観点から、「数理的

手法」、「ヒューリスティックな手法」および「確率的な最適化手法」に分類できる。各手法はそれぞれ一長一短があるが、ここでは本論文に関連する各手法の問題点について述べる。

- (1) 数理的な手法： 提案されているアルゴリズム^{5),6)}は、配置要素を矩形状に限定し、非矩形状を対象としていない。そのため、実際の生産現場で対象となる複雑な形状をした配置要素に効率的に対応できないという問題点がある。
- (2) ヒューリスティックな手法： Quらは配置要素を4種類の非矩形状に限定した非矩形状の板取システムを開発した⁷⁾。また、Cheokらは、配置要素を船舶部品に限定した板取システムを開発した⁸⁾。彼らは、船舶部品の特徴から配置要素を単純な矩形状に近似することで(凹形状の部分を埋める操作)、非矩形状配置問題を矩形状配置問題として処理し、実用的なシステムを構築した。これらの方法では、各配置要素1つ1つに対し配置手続きを用意する必要がある、対象外の配置要素に対しては、その形状を最大矩形で近似して配置を行う必要がある。そのため、無駄となる材料(スクラップ量)が多くなるという問題点がある。
- (3) 確率的な最適化手法： JainらのSA(Simulated Annealing)による方法⁹⁾や藤田らのGA(Genetic Algorithm)による方法¹⁰⁾では、基本的には配置要素の形状に依存していない。しかし、これらの方法では、配置要素間の重なり計算などを精密に行わなければならない、事前に配置要素の形状に基づいた計算手続きを用意する必要がある。また、過去に板取を行った配置要素とそうでない要素との区別をしないため、過去に板取を行った要素に対して、過去の板取結果を利用できない。すなわち、一度板取を行った要素群に新たに要素を1つ追加しただけでも、再度すべての配置要素に対して配置計画を行うため、処理に多くの時間を要するという問題点がある。

上記の問題点を改善するために、本研究では次の3つのことを考慮する。

- 非矩形状への対応
- 新規の配置要素への対応
- 過去の板取結果の利用

本研究では、配置要素の形状特徴に基づいた、同一の要素を2つ以上組み合わせたときに効率が良い配置方法に従って、配置要素をタイプ分類する。よって、局所的に各要素ごとに配置計画を行う。これにより、配置対象数を実質的に削減することができる。また、

タイプ分類の前処理として配置要素の形状認識を行うが、その方法として、過去の板取事例の利用や形状近似を使う。形状近似により、非矩形の配置要素をタイプ分類で想定した形状として扱うことができ、また、新規の配置要素に対しては最大矩形領域より少ない領域の形状として処理できる。過去の板取事例の利用は、事例ベース推論¹¹⁾によって実現する。

3. 部品の形状特徴に基づく板取システムの概要

3.1 板取の前提条件

NC加工分野における板取問題では、数学分野で扱われる一般的な幾何学図形の配置問題のように、最適解を求めただけでは解決したことにならない。換言すると、板取問題においては、最適性のある程度犠牲にすることで、より優先しなければならない制約が存在する。たとえば、配置方法によって材料の強度が不足して製品の機能を満足できなくなってしまうことや、処理に要する時間が実用的であることなどある。このようなことをふまえて、本研究では、以下の前提条件の下で配置方法を考案する。

- 配置する部品は、金属加工の性質や金属の目を考慮して、 ± 90 度回転、軸対称の反転のみ可能とする。
- 加工時の効率を上げるため、同一形状の部品はできるだけ近くに配置する。なお、システムでは、同一形状の部品を隣接して配置することにより実装を行う。

ただし、金型で材料をパンチしたときに生じる様々な影響や製品の強度などは、現時点では考えないものとする。

また板取問題では、配置計画を行う際にどのような要件を重視するのかによって、配置方針が変わってくる。代表的な要件として以下のものがあげられる。

- (1) 製品の単位は崩さずに、NC加工後の作業（仕分け、組立て、在庫管理、製品ごとの曲げ加工など）の効率を考慮して配置すること。
- (2) 製品の単位は崩しても、できるだけ多くの部品を材料の無駄がないように効率良く配置すること。

本研究では、(1)よりも(2)の要件を優先した配置方針を採用する。

3.2 システム構成

本システムは、CAD/CAMシステムにおける工程設計の一部分の支援を行うことを目的としている。そのため、本システムへの入力にはCADの出力データと

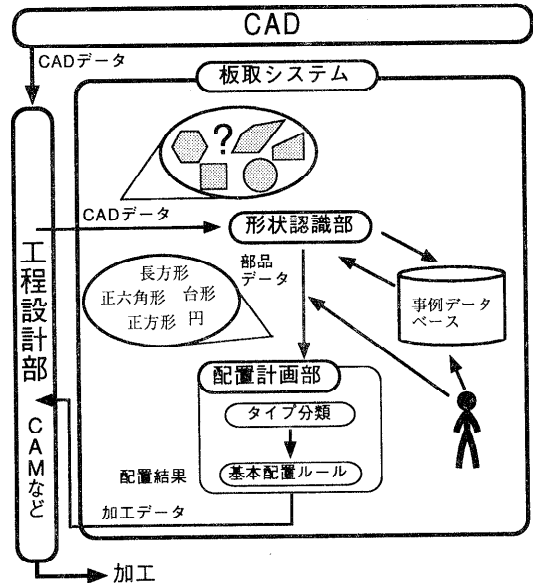


図1 システム構成

Fig.1 Configuration of the system.

し、それに処理を施すことで配置計画を行い、結果を形状データとして出力し、次のCAMの段階、たとえばNCデータ生成システムへと受け渡す形態を採用した。図1に本システムの構成を示す。

本システムで想定しているCADデータは、配置要素を線分、円および円弧といった図形要素で表現しており、形状として表現しているわけではない。そのため、まず形状認識部で各要素の形状を認識し、形状の情報を加えたデータを配置計画部に渡す。配置計画部では、形状に基づいて配置要素をタイプ分類し、分類された要素を基本配置ルールに従って板材料上に配置する。この結果として得られたデータを工程設計部に渡し、その他の工程設計のデータと合わせて実際の加工を実施することになる。たとえば著者らの開発したNC加工プログラム自動生成システム¹⁴⁾を利用して加工シミュレーションを行う。

3.3 形状認識部

配置計画部でのタイプ分類の前処理として、CADデータから配置要素の形状を認識する必要がある。その方法として、事例ベース推論を使う。入力されたCADデータに対し、過去に配置した要素のCADデータ（事例データベース）の中から、その形状特徴をインデックスとして最も類似しているものを検索し、形状を決定する。新規の配置要素に対しては、その形状を事例データベースにすでに登録されている形状に近似することで対応する。

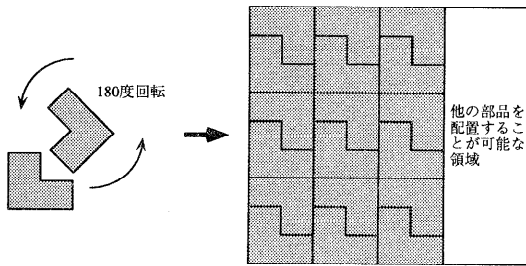


図2 L字型形状の部品の配置
Fig. 2 Layout of L-shape parts.

3.4 形状特徴による配置計画部

NCT 加工では、1枚の金属板材からまったく異なる形状の部品を多数切り取る場合は少なく、同一形状の部品を多数切り取る場合が多い。そのため、配置対象として同一形状の部品が複数存在する場合、それぞれを別々の配置要素としてとらえるのではなく、部品の形状特徴に基づいて複数の部品をまとめて1つの配置要素として処理する方が効率的である。たとえば、配置要素としてL字型の形状をした部品が複数ある場合、図2のように、一方の部品を180度回転させ、もう一方の部品と組み合わせて配置を行う。ただし、配置要素が奇数個の場合には、組を作れない1個の配置要素に対してはその最大矩形によって配置を行う。こうすることで、2つのL字型の配置要素を1つの矩形の配置要素として扱うことが可能となり、配置要素を減少できる。また、このような部品の形状特徴によって部品を分類（タイプ分類）し、各分類に基づいた配置方法によって各部品ごと局所的な配置計画を行う。その後、各部品群（タイプ分類後の部品）に対する大局的な配置計画を行う。こうすることで、従来のシステムのように配置要素すべてを対象とする大局的な配置計画を回避でき、配置計画に要する時間が短縮できる。配置計画部では、このような部品の形状特徴によって部品を分類し、その分類に基づいて部品を板材に配置する。

4. 部品の形状認識法

図3に形状認識部の処理の流れを示す。CADデータは部品の各頂点座標データであり、その座標データから部品の形状特徴を求め、形状認識を行う。ただし、ここでの形状認識とは、部品の形状名を決定することである。形状認識の方法として、過去の事例を参照する事例ベース推論を使う。事例ベース推論に失敗した場合、形状近似部で形状の凹凸部を矩形で近似し^{*}、事例ベース推論でその形状名を決定する。形状近似後さらに事例ベース推論による形状名の決定に失敗し

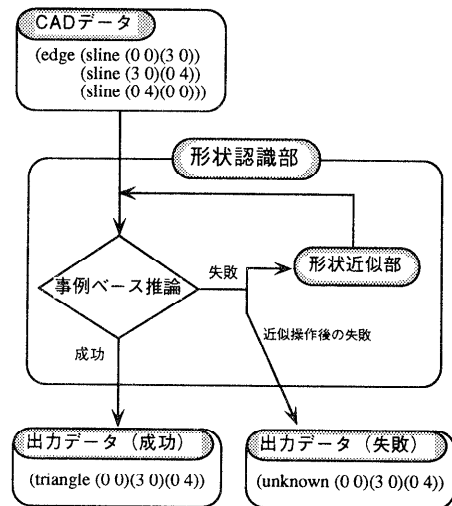


図3 形状認識部の処理の流れ
Fig. 3 Procedure of shape recognition.

た場合、形状をその最大矩形で近似し、“unknown”という仮の形状名を割り付ける。したがって、この“unknown”の要素を配置する際には、元の形状の最大矩形領域が必要となり、その結果、スクラップ量を増加させることになる。そのスクラップ量をできるだけ少なくするために、ユーザによって上記の最大矩形より小さい矩形で近似し、新たな形状名を割り付ける。この近似した形状は、新規事例として、近似前の形状や近似後の形状を含め、事例データベースに追加する。また、ユーザによって近似が行われない場合には、その“unknown”の形状は、失敗事例として事例データベースに追加される。

このような失敗事例とマッチした配置要素が入力された場合には、システムがユーザに対し、失敗事例とマッチしたことを通知し、ユーザによる形状近似を促す。ただし、事例を元のデータベースと矛盾することなく追加することは容易でないため、本システムではその矛盾に関係なく新規事例を事例データベースに追加することを許している。その代わりに事例ベース推論で複数の事例がマッチした場合には、ユーザはシステムとの会話的操作によってマッチした事例の内容を把握し、どちらの事例を採用するかを決定する。

4.1 事例の利用

本システムの事例ベース推論では、初めに配置要素のCADデータから、その形状特徴を示すインデックス値を生成する。そのインデックスと事例データベー

^{*} なお、現在、近似対象として、線分要素で構成される配置要素に限定している。

スに登録された事例（過去に配置した要素に関するもの）のインデックスとを4.1.3項で述べる方法で照合し、最も類似している事例を選び出す。選ばれた事例のNAMEインデックス値（形状名）が、入力されたCADデータの形状名であるとし、形状を認識したことにする。

4.1.1 事例のインデックス

インデックスとしては、頂点数、斜辺の数、平行線分の数および、配置要素の各辺の方向性を用いる。各辺の方向性とは部品の各頂点間の辺の傾きに該当し、その表現法は各頂点間のX軸成分およびY軸成分の変化を反時計まわりで「増加」、「一定」および「減少」の3つの状態で表現する。事例データベースには、「増加」、「一定」および「減少」をそれぞれ、「+」、「0」および「-」で表現する。たとえば部品の中の2つの頂点A（座標値が(3, 4)）と頂点B（座標値が(5, 2)）間の辺の方向性は、X軸成分の変化が「増加」でY軸成分の変化が「減少」であるため、「(+ -)」と表現する。同様に、4つの頂点座標(0, 0), (10, 0), (10, 50)および(0, 50)を持つ部品の場合、その各辺の方向性の表現は「((+ 0)(0 +)(- 0)(0 -))」となる。

4.1.2 事例データベース

事例データベースの記述例を図4に示す。同図は、三角形と四角形に関する事例である。同図で、AXIS, AXIS-NUM, OBLIQUE-SIDE, PARALLEL-LINEおよびVECTORは、各々配置要素の頂点座標、頂点数、斜辺の数、平行線分の数および各辺の方向性を示すインデックスである。これらのインデックスの値は、本システムのCADデータから、以下に示す手続きによって自動的に生成される。

- 頂点座標値 (AXIS) : CADデータに対するリスト処理で求める。
- 頂点数 (AXIS-NUM) : 頂点座標値のリストの長さで求める（具体的には、Common Lispの「length」コマンドを使う）。
- 斜辺の数 (OBLIQUE-SIDE) : Step-1 : 各辺の傾きを求め、Step-2 : 傾きが異なるものがいくつあるかをカウントし、その値が斜辺の数である。
- 平行線分の数 (PARALLEL-LINE) : Step-1 : 各辺の傾きを求め、Step-2 : 傾きが同じものがいくつあるかをカウントし、その値が平行線分の数である。
- 各辺の方向性 (VECTOR) : 反時計まわりに、各頂点間のX軸成分およびY軸成分の変化を計算し、「増加」、「一定」および「減少」の3つの状態で示し、各状態を「+」、「0」および「-」でラベ

```
(( (AXIS ((0 0) (8 2) (0 8)))
  (AXIS-NUM 3)
  (OBLIQUE-SIDE 2)
  (PARALLEL-LINE 0)
  (VECTOR ((+ +) (- +) (0 -)))
  (MATCHING-COUNTER 0)
  (NAME TRIANGLE)
  (ARRANGEMENT-TYPE TYPE-2))
(( (AXIS ((0 0) (4 0) (4 2) (0 2)))
  (AXIS-NUM 4)
  (OBLIQUE-SIDE 0)
  (PARALLEL-LINE 2)
  (VECTOR ((+ 0) (0 +) (- 0) (0 -)))
  (MATCHING-COUNTER 0)
  (NAME RECTANGLE)
  (ARRANGEMENT-TYPE TYPE-1))
...)
```

図4 事例データベースの例（三角形および四角形の例）

Fig. 4 Example of case database (triangle, rectangle).

ル付ける。

NAMEは部品の形状名を示すインデックスで、事例ベース推論時に決定される形状名が格納される。ARRANGEMENT-TYPEは5.1節で示す配置方法のタイプである。たとえば、ある配置要素が三角形と認識された場合、その配置方法はTYPE-2として決定される。MATCHING-COUNTERは、事例が選択された回数を記録するためのインデックスである。この他に、形状近似部での形状近似法1~3を適用した結果の形状情報（頂点座標）を格納するインデックスや失敗事例であることを示すインデックスがある。

4.1.3 事例の検索

入力されたCADデータから、4.1.1項で述べた座標の数、斜辺の数、平行線分の数、各直線の方向性に関するインデックスの値をシステムが自動的に算出する。そして、各インデックスに関して、事例データベースに格納された事例との類似距離を調べる。この場合の類似距離は事例との差で表現する。たとえば座標の数が3の部品と座標の数が4の事例との類似距離は1ということになる。つまり、類似距離が小さいほど、類似した事例ということになる。システムでは各インデックスの類似距離の合計で、最も小さい値の事例を採用することとした。なお、必要十分なインデックスの種類や類似距離の算出法を一般に決定するのは困難であるため、現在のシステムでは、上記の4つのインデックスを使用している。そして、それらのインデックスと完全に一致した事例を採用する方法で、本論文で述べる実験において事例ベース推論の有効性が確認できた（詳細は、6.2節で説明）。

4.1.4 新規事例の作成

本システムでは、配置要素の形状を認識することで、その配置方法を決定する。そこで、形状を認識できなかった要素(図3の出力データ(失敗))は“unknown”という仮の名称をつけて配置計画部に渡す。配置計画部では、“unknown”という形状の要素はその最大矩形で配置するため(5.1節で述べるType-5として扱う)、スクラップ量を多く含んだ形状で処理されることになる。そのため、ユーザがその配置方法を(1)システム実行中に5.1節で述べるTypeをユーザが指定する;(2)システムを一度終了させ、上記の最大矩形より小さい矩形で形状近似を行いTypeをユーザが指定する、という手順で検討して、その配置方法、近似した形状とともに新たな形状名をつけて、新規事例として事例データベースに追加する。

4.2 配置要素の形状近似法

NCTによって実際に加工される部品の形状は、長方形や三角形のような単純な形状ではなく、凹凸のあるものが多い。また、本システムの事例ベース推論による形状認識方法では比較的単純な形状を認識することはできるが、すべての形状を認識できるわけではない。しかし、それらが単純な形状とかけ離れているものばかりではなく、たとえば、切り欠きなどを無視して大局的に見れば単純な形状に近似可能なものが多い。そこで、凹凸のある配置要素を単純な形状に近似することで、事例ベース推論による形状認識の補完を行う。本システムでは、線分要素で構成された配置要素の形状近似方法として、以下の3種類を用意した。これらの近似方法を組み合わせて近似を行う。なお、配置要素の凹凸部分を検出するには、頂点座標が内部点か外部点かを特定する必要がある。そのために、付録A.1に示す手続きを用いた。以下の説明で、 P_1, P_2, P_n とはそれぞれ頂点1, 2, n のことを意味する。

(1) 形状近似法1 配置要素中の凹の部分埋める操作を行う¹²⁾。図5に形状近似法1の例を示す。

Step-1: 最初にすべての頂点に対し3つの頂点からなる組 $((P_1 P_2 P_3)(P_2 P_3 P_4), \dots, (P_6 P_1 P_2))$ を生成する。

Step-2: 任意の組(以降、 $(P_4 P_5 P_6)$ の組について説明する)の両端の頂点(P_4 と P_6)を線分で結ぶ。その線分が両端の頂点以外と配置要素の内部と交差する(接する場合は除く)かを調べる。

Step-3: 交差する場合は近似処理を行わない。交差しない場合(接する場合を含む)、3つ頂点の組のうち、2番目の頂点(ここでは P_5

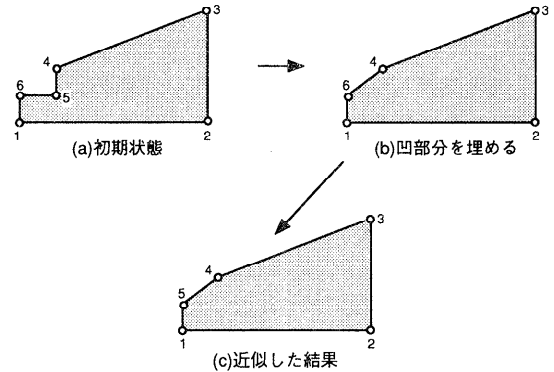


図5 形状近似法1の例

Fig. 5 Example of shape approximation method 1.

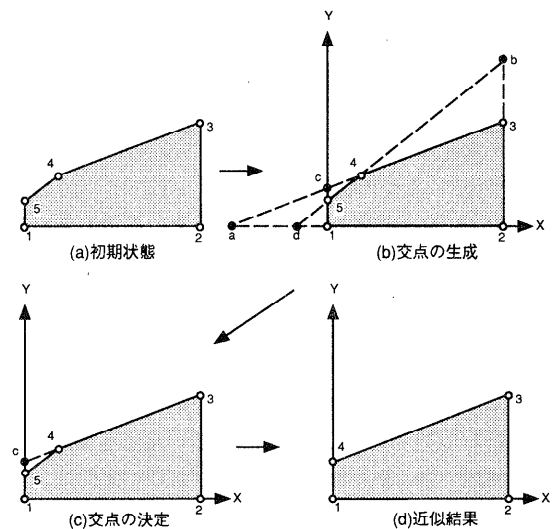


図6 形状近似法2の例

Fig. 6 Example of shape approximation method 2.

のこと)を削除する。

Step-4: Step-2からStep-3の処理を、すべての頂点の組に対し行う。

Step-5: 最終的に頂点番号の再割付けを行う(同図(c))。

(2) 形状近似法2 凸形状をした配置要素に対する近似操作である。たとえば、図6(a)のような凸形状を図6(d)のように面積を追加して近似を行う。

Step-1: 最初にすべての頂点に対し4つ頂点からなる組 $((P_1 P_2 P_3 P_4)(P_2 P_3 P_4 P_5), \dots, (P_5 P_1 P_2 P_3))$ を生成する。

Step-2: 任意の組(以降、 $(P_1 P_2 P_3 P_4)$ の組について説明する)の前半の2頂点(P_1 と P_2)を通る直線と、後半の2頂点(P_3 と P_4)を通る直線の交点(ここではaのこと)

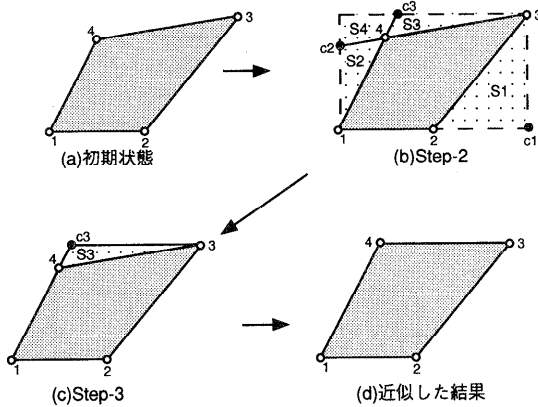


図7 形状近似法3の例

Fig. 7 Example of shape approximation method 3.

を求める。この交点は近似してできる新しい形状の頂点の候補である。

Step-3: Step-1で生成した頂点の組に対して、Step-2の操作を行う(同図(b)の交点a~dが求まる)。

Step-4: 各交点によってできる新しい形状の領域のうち、最小かつ元の形状の最大矩形領域より小さい場合、そのときの交点を新たな頂点として採用する(同図(c)で交点c)。

Step-5: 最終的に頂点番号の再割付けを行う(同図(d))。

(3) **形状近似法3** 配置要素が非矩形形状の四角形の場合、その最大矩形をとることを回避する近似操作である。たとえば、図7(a)のように形状近似法2ではこれ以上近似操作を行えない場合、図7のように面積を追加して近似を行う。

Step-1: 四辺の傾きを求める。底辺(同図の2頂点(P_1 と P_2)間の辺)以外の辺の傾きの正負によって場合分けをする。

Step-2: 元の形状の最大矩形領域(同図(b)の点線の部分)と四辺を延長した直線との交点を求める(同図(b)の c_1 , c_2 と c_3)。この交点は近似してできる新しい形状の頂点の候補となる。ただし、求めた交点が最大矩形領域の各頂点と一致する場合には、その交点は無視する。

Step-3: 各交点によってできる新しい形状の領域(同図(b)の S_1 , S_2 , S_3 , S_4)のうち最小であるもの^{*}(ここでは S_3 のこと)、その時の交点(ここでは c_3 のこと)が新たな頂点として採用される。

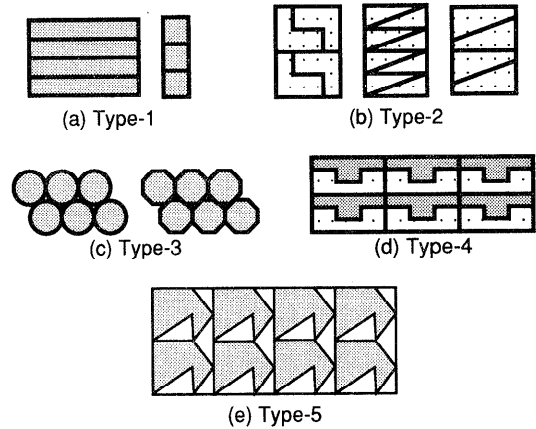


図8 タイプ分類

Fig. 8 Type classification.

Step-4: 最終的に頂点番号の再割付けを行う(同図(d))。

5. 配置計画法

部品の形状特徴に基づいてタイプ分類し、基本配置ルールに従って配置計画を行う。

5.1 タイプ分類

様々な部品を効率良く配置するために、部品をその形状特徴(具体的には、事例データベースのARRANGEMENT-TYPE)から次の5つのタイプにシステムが分類する¹³⁾。その例を図8に示す。ただし、形状認識に失敗した部品に対しては、ユーザがタイプ分類を行う。

- (a) **Type-1:** 長方形や正方形などが含まれる。配置するときはそれぞれの部品を同じ状態(向きなど)にしておき、図8(a)のように配置する。
- (b) **Type-2:** 台形、三角形、L字型などが含まれる。これらの形状の部品は2つを組にして、一方を180度回転させて他方の上に重ねるように配置する。これらは2つを組み合わせることにより、Type-1に分類される形状と同じになり無駄な部分なくなる(図8(b))。
- (c) **Type-3:** 円や正八角形など円形に近い部品が含まれる。たとえば、1列目をY軸方向に重ねて配置した場合、2列目をY軸方向に、たとえば

^{*} Step-1の場合分けにおいて、頂点 P_2 と P_3 間の辺の傾きと頂点 P_4 と P_1 間の辺の傾きが正、頂点 P_2 と P_3 間の辺が負の場合、交点によってできた新しい領域のうち最小なものを選択しただけでは、5.1節で述べるタイプ分類による配置を活用できない場合がある。その場合には新しい領域の組合せによって近似を行う。

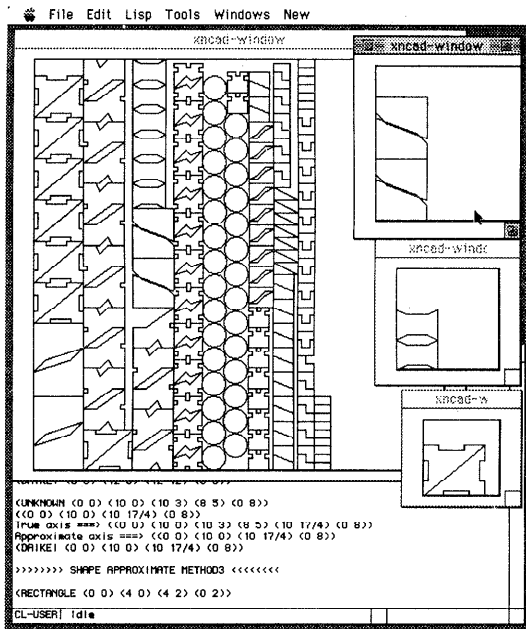


図9 プロトタイプシステム

Fig. 9 Display image of our system.

円であれば、半径分だけずらして重ねて配置する（図8(c)）。

(d) **Type-4**: 凹凸型のように2つは異なった形状の部品であるが、組み合わせることによって無駄が少ないように配置する（図8(d)）。

(e) **Type-5**: Type-1~4に当てはまらなかった形状の部品で、これらは従来のように最大矩形をとって配置する（図8(e)）。

以上のように分類することにより、各タイプの同一形状の部品は無駄をできるだけ少なくした形でまとめて配置することができる。これは従来のように最大矩形をとったときに生じた矩形内に含まれる無駄の分を少なくすることになる。

5.2 基本配置ルール

各タイプに分類された部品を配置するとき、通常、熟練者が行っているように部品の面積の大きい順に配置を行う。このとき配置の方向は、NC加工における原点である材料の最も左下を原点とし、材料（配置領域）に対し縦方向と横方向の2方向について行い、両者のうち配置効率の良い方を最終的に採用する。

6. システムの実行情例

6.1 プロトタイプシステム*

これまでに述べてきた部品の形状認識法・配置計画法に基づいて、その有効性を検証するためにプロトタイプシステムを作成した（図9）。システムの開発は

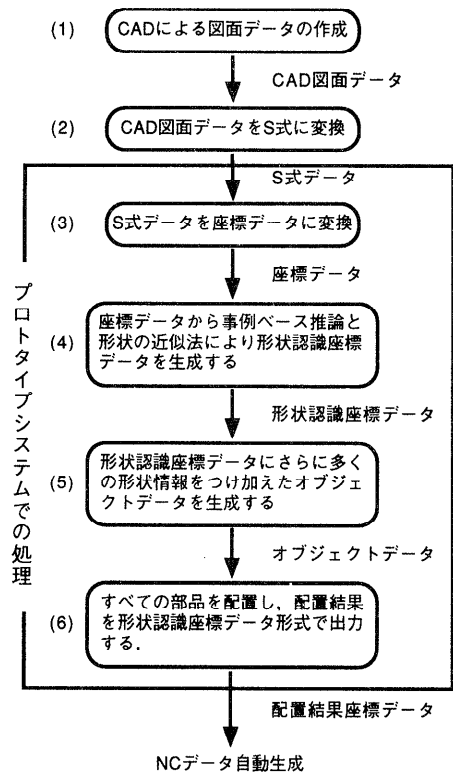


図10 プロトタイプシステムのデータ処理の流れ
Fig. 10 Flow diagram of prototype system.

CLOS (Common Lisp Object System) を用いた。

プロトタイプシステムでのデータ処理の流れを図10に示す。同図で(1)から(2)では、筆者らが開発したNC加工プログラム自動生成システム¹⁴⁾や設計支援システム¹⁵⁾とのデータの互換性を確保するため、S式に変換している。ただし、この変換は先のシステムで行い、本システムでは(2)のデータ形式が入力データとなる。次に、(2)のデータ形式を(3)の座標値形式に変換する。(3)からその形状を認識するために、事例ベース推論や形状の近似を行うことによって(4)を作成する。さらに、(4)に形状に関するその他の情報を加えた(5)を生成し、(5)を参照し実際に配置していくことによって、最終的に(6)を出力する。付録A.2に各データ形式を示し、付録A.3にプロトタイプシステムの実行過程の例を示す。

6.2 システムの実験結果

本手法の局所的な配置計画の有効性を検討するために、NCTによって加工されるラックの構成部品の一部（非矩形形状を含む15種類の配置要素、総個数212

* 近似対象として線分要素で構成される配置要素に限定している点からプロトタイプシステムとした。

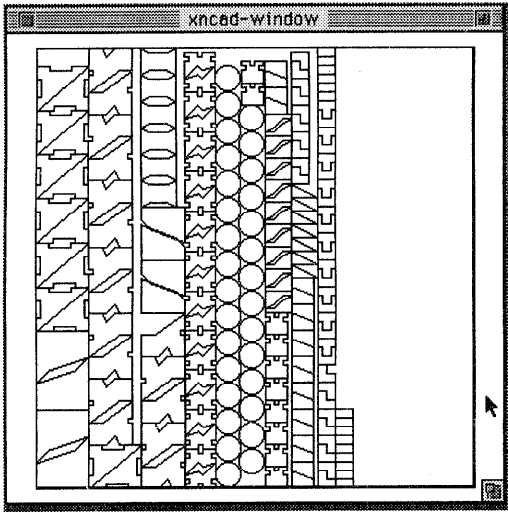


図 11 本手法による実行結果 (Case-1)
Fig. 11 Nesting result by our system.

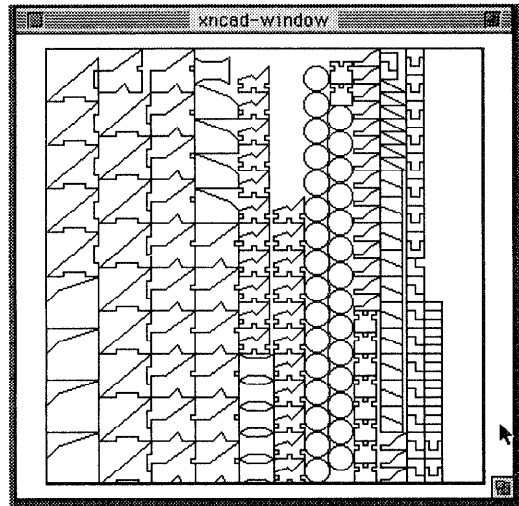


図 12 形状近似を適用しない場合の実行結果 (Case-2)
Fig. 12 Nesting result without approximating parts.

個) に対し, 事例データベースの初期状態として 70 個の事例を使い, 以下の 3 つのケースについて実験を行った。ただし, 各実験のたびに, 配置要素はすべて新規の配置要素としてシステムに入力し, 事例データベースを初期状態にした。また, 形状近似法の後に事例ベース推論を行った結果, 形状認識できない配置要素に対し, 各実験間の処理時間を比較するために, ユーザによる会話的操作を行わずに, “unknown” として処理しその最大矩形で配置計画を行った。

Case-1: 本システムの手法 (形状近似およびタイプ分類を使う) による配置。

Case-2: 形状近似法を使わず, タイプ分類のみで配置。

Case-3: 形状近似法およびタイプ分類を使わず, すべての要素に対して最大矩形領域で配置。

それぞれの配置結果を図 11, 図 12, 図 13 に示す。3 者について板取に必要な領域*および処理時間**を比較し, その結果を表 1 に示す。

板取に必要な領域に関しては, 本手法 (Case-1) が他の方法 (Case-2, 3) よりも少ない領域で配置可能となった。この理由は, (1) 形状近似によって非矩形の配置要素をタイプ分類で想定した形状として扱うことができ, 同一種類の配置要素に対する局所的な配置計画を行った (総個数 212 個の配置要素を 138 個の配置

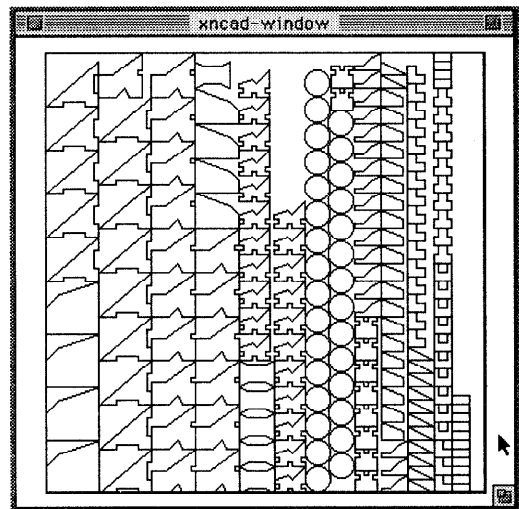


図 13 すべての配置要素の最大矩形を取った場合の実行結果 (Case-3)
Fig. 13 Nesting result by maximum rectangular area of parts.

要素として処理した); (2) 表 1 の Case-1 と Case-2 での板取に必要な領域の割合の差から, 形状近似法によって, 新規の配置要素に対して最大矩形領域より少ない領域の形状として処理できたためである。

一方, 処理時間に関しては, Case-1 では形状近似の分だけ, 処理時間が多くかった。また, Case-2 と Case-3 の処理時間がほぼ等しかった。その理由は, Case-3 の場合, 実質的にすべての配置要素をタイプ分類の Type-5 として処理し, 実験を行ったためである。

さらに, 事例ベース推論の有効性を調べるために,

* ここでの比較方法としては, 配置領域の板材全体に対する割合を用いた。ただし配置領域とは配置した領域の最大矩形をとった。

** Power Macintosh 8500/120 (CPU: PowerPC604 120 MHz) 上で実行したものである。処理時間の計測には, Common Lisp の time コマンドを使用した。

表 1 実験結果

Table 1 Results of experiments.

実験	板取に必要な領域の割合	形状認識部での処理時間	配置計画部での処理時間	合計処理時間
Case-1	約 72%	約 7.5 秒	約 20.3 秒	約 27.8 秒
Case-2	約 90%	約 1.1 秒	約 20.2 秒	約 21.3 秒
Case-3	約 96%	約 1.1 秒	約 20.3 秒	約 21.4 秒
Case-1'	約 72%	約 1.2 秒	約 20.5 秒	約 21.7 秒

処理時間は小数点以下第 2 位で四捨五入をした。

以下の実験を行った。

Case-1'： Case-1 の実験を行った後、事例データベースを初期状態に戻さずに、再度同じ配置要素に対し板取を行った。

表 1 に示すように、Case-1' では Case-1 に比べて形状認識部の処理時間が短縮した。これは、事例データベースに格納された過去の配置事例によって形状近似処理が省略されたためである。したがって、この例では、事例ベース推論を利用することが有効であることが確認された。

7. む す び

本研究では、配置要素の形状特徴に基づいた NC 加工の板取システムを提案し、実際にプロトタイプシステムを用いて、その有効性を示した。本手法では、形状特徴に基づいた、同一の要素を 2 つ以上組み合わせたときに効率が良い配置方法に従って配置要素をタイプ分類する。これにより、同一種類の配置要素に対する局所的な配置計画を行う。このような配置計画によって、大域的な配置計画での配置対象数を実質的に削減することができた。また、タイプ分類の前処理として配置要素の形状認識を行うが、その方法として事例ベース推論による過去の板取事例の利用や形状近似を使った。過去の事例によって、再度板取を行う際に形状近似を行わずに形状認識でき、局所的な配置計画を行うことができた。形状近似により、非矩形の配置要素をタイプ分類で想定した形状として扱うことができ、また、新規の配置要素に対しては最大矩形領域より少ない領域の形状として処理できた。

今後の課題としては、以下のことがあげられる。

- 事例データベースのインデックスの項目について、さらに多くの実験を通して検討を加え、本システムの事例ベース推論の性能評価を行うこと。
- 現在の形状近似法では、線分要素で構成される配置要素に限定しているが、円形の配置要素および円弧を含む配置要素にも対応すること。
- 本手法では基本的に同一種類の配置要素に対する

局所的な配置計画を中心としたものである。今後は、大域的な配置計画部である基本配置ルールの検討を行い、SA (Simulated Annealing) や GA (Genetic Algorithm) のような最適化手法の導入を検討すること。

謝辞 本研究を行うにあたり、(株) MI ベンチャーズの猪飼國夫氏より、配置事例データおよび部品データ等を快くご提供いただきました。ここに感謝の意を表します。

参 考 文 献

- 1) 藤田喜久雄, 赤木新介, 土居秀人: プラント配置設計における配置制約の自動生成, 日本機械学会論文集 (C), Vol.58, No.555, pp.3449-3454 (1992).
- 2) Welgama, P.S. and Gibson, O.R.: An integrated methodology for automating the determination of layout and materials handling system, *International Journal of Production Research*, Vol.34, No.8, pp.2247-2564 (1996).
- 3) Sweeney, P.E. and Paternoster, E.R.: Cutting and packing problems: A categorized, application-oriented research, bibliography, *Journal of the Operational Research Society*, Vol.43, No.7, pp.691-706 (1992).
- 4) 穂坂 衛, 佐田登志夫: 統合化 CAD/CAM システム, pp.331-404, オーム社 (1994).
- 5) Baker, B.S. and Schwarz, J.S.: Self algorithms for two-dimensional packing problems, *SIAM J. Comput.*, Vol.12, No.3, pp.508-525 (1983).
- 6) Coffman, E.G. Jr. and Shor, P.W.: Average-case analysis of cutting and packing in two dimensions, *European Journal of Operational Research*, Vol.44, pp.134-144 (1990).
- 7) Qu, W. and Sanders, J.L.: A nesting algorithm for irregular parts and factors affecting trim losses, *Journal of the Operational Research Society*, Vol.43, No.7, pp.691-706 (1987).
- 8) Cheok, B.T. and Nee, A.Y.C.: Algorithms for nesting of ship/offshore structure plates, *Advances in Design Automation*, ED-Vol.32,

No.2, pp.221-226 (1991).

- 9) Jain, P., Fenyas, P. and Richter, R.: Optimal blank nesting using simulated annealing, *Advances in Design Automation*, DE-Vol.32, No.2, pp.109-116 (1990).
- 10) 藤田喜久雄, 赤木新介, 廣川敬康: 遺伝的アルゴリズムと極小値探索アルゴリズムとのハイブリット化による板取問題の一解法, 日本機械学会論文集 (C), Vol.59, No.564, pp.312-319 (1994).
- 11) 松原 仁: 推論技術の観点からみた事例に基づく推論, 人工知能学会誌, Vol.7, No.4, pp.11-19 (1992).
- 12) 田中功二, 若林伸和, 金井秀明, 本多中二: NC機械工作における部品の形状配置法の提案, 第48回情報処理学会全国大会論文集, Vol.6, pp.183-184 (1994).
- 13) 田中功二, 金井秀明, 若林伸和: NC加工における部品の形状配置問題に関する研究, 情報処理学会グラフィックとCADシンポジウム, pp.135-142 (1993).
- 14) 若林伸和, 北橋忠宏: パターン認識の手法を利用したNCパンチプレス加工プログラム自動生成システム, 電気学会論文誌 (D), Vol.113-D, No.6, pp.778-785 (1993).
- 15) 金井秀明, 若林伸和, 本多中二: 設計の上流工程を考慮した設計・製造プロセス統合支援システム, 情報処理学会論文誌, Vol.36, No.8, pp.1926-1937 (1995).

付 録

A.1 配置要素の座標値の内部・外部の定義

線分要素から構成される配置要素の座標点列を $(P_1, P_2, \dots, P_n)$ とする。ただし、この列は原点から反時計まわりの順である。この列の任意の点 P_l が配置要素の内部・外部かを式 (1) で求める D の値によって判定する。 $D < 0$ のとき、 P_l は内部、 $D > 0$ のとき、 P_l は外部と判定する。ここで、点 P_l の X , Y 座標を x_l , y_l と表現する。

$$D = (y_{l+1} - y_{l-1})(x_l - x_{l-1}) - (x_{l+1} - x_{l-1})(y_l - y_{l-1}) \quad (1)$$

ただし、 $1 \leq l \leq n$, $x_{n+1} = x_1$, $y_{n+1} = y_1$, $x_0 = x_n$, $y_0 = y_n$ である。

A.2 プロトタイプシステムでのデータ形式

A.2.1 CAD図面データ形式

CAD図面データ形式は著者らが開発したCADのデータ形式で、表2に示すコマンド群によって図面を表現する。

A.2.2 S式データ形式

S式データ形式を表3、表4に示す。これはCAD図面データをS式に変換したものである。ただし、次

表2 CAD図面データ形式

Table 2 CAD data format.

コマンド	内容	形式
WORK	材料サイズ	X Y
RANGE	利用する材料の範囲	RANGE Xs Ys Xe Ye
SLINE	線分	SLINE Xs Ys Xe Ye
CIRCLE	円	CIRCLE Xc Yc R
ARC	円弧	ARC Xc Yc R Xs Ys Xe Ye

Xs: 開始点のX座標値, Ys: 開始点のY座標値, Xe: 終了点のX座標値, Ye: 終了点のY座標値, Xc: 中心点のX座標値, Yc: 中心点のY座標値, R: 半径, X: X方向の長さ, Y: Y方向の長さ

表3 S式データ (図形の種類)

Table 3 Shape in S-format.

コマンド	内容	形式
WORK	材料のサイズ	(WORK (X Y))
EDGE	材料の外縁	(EDGE (要素)... (要素))
HOLE	穴開け加工	(HOLE (要素)... (要素))

X: X方向の長さ, Y: Y方向の長さ

表4 S式データ (要素)

Table 4 Element in S-format.

コマンド	内容	形式
SLINE	線分	(SLINE (Xs Ys) (Xe Ye))
CIRCLE	円	(CIRCLE (Xc Yc) R)
ARC	円弧	(ARC (Xs Ys) (Xe Ye) (Xc Yc) R)

Xs: 開始点のX座標値, Ys: 開始点のY座標値, Xe: 終了点のX座標値, Ye: 終了点のY座標値, Xc: 中心点のX座標値, Yc: 中心点のY座標値, R: 半径

のような変換規則に従うものとする。

- 1つの図形 (穴または外縁) を1つのリストとし、それを線分、円、円弧という図形の構成要素の組合せで表現する。
- 1つの図形はYが最小かつXが最小である点から開始し、反時計まわりで連続した構成要素の並びとする。

A.2.3 座標データ形式

座標データ形式はS式データをそれが表す図形の各頂点座標で表したもので、((X0 Y0)(X1 Y1) ... (Xn Yn)) のように座標値のリスト形式である。この座標の並び方は、S式データと同様にYが最小かつXが最小である点から開始し、反時計まわりで連続した構成要素の並びとする。

A.2.4 形状認識座標データ

形状認識座標データ形式を表5に示す。このデータは座標データに形状の情報を加えた形状認識座標データを表す。形状の情報は、事例ベース推論および形状の近似により得られる。新しい形状の部品が与えられたときには、それを新規事例として追加することがで

表5 形状の種類の例
Table 5 Example of shape.

形状名	内容	形式
rectangle	長方形	(rectangle (X0 Y0)(X1 Y1) (X2 Y2)(X3 Y3))
triangle	三角形	(triangle (X0 Y0)(X1 Y1) (X2 Y2))
ltype	L字型	(ltype (X0 Y0)(X1 Y1) ... (X5 Y5))
circle	円	(circle (Xc Yc) R)
unknown	未知形状	(unknown (X0 Y0)(X1 Y1) ... (Xn Yn))

表6 オブジェクトデータの例 (台形)
Table 6 Example of object data format.

属性	属性値
座標	((0 0)(12 0)(12 9)(0 9/2))
近似前座標	((0 0)(12 0)(12 9)(4 6)(4 3)(0 3))
名称	DAIKEI
縦辺長	9
横辺長	12
配置方法	type-1

きる。

A.2.5 オブジェクトデータ

板材料に部品を配置していくときに、実際に操作しなければならないのは座標である。システムはオブジェクト指向により記述しているので、すべての形状認識データはオブジェクトとして扱われる。このオブジェクトデータは形状認識データに、さらに多くの形状情報を加えたデータである。情報を加えることによって、実際に部品を板材料に配置するときに必要な処理がしやすくなる。表6に示す。

A.2.6 配置結果データ形式

配置結果データ形式はオブジェクトデータを形状配置システムにより配置した結果のデータを表したものである。このデータには部品の形状と座標が含まれている。

A.3 プロトタイプシステムの処理の例

図5の部品に対して、プロトタイプシステムを実行した処理過程を図14に示す。

(平成9年3月10日受付)

(平成9年9月10日採録)

金井 秀明 (正会員)



1969年生。1993年電気通信大学大学院電子情報学専攻博士前期課程修了。1996年同博士後期課程単位取得退学。同年、同大学大学院情報システム学研究科助手、現在に至る。設計支援システム等の情報システムの研究に従事。計測自動制御学会、人工知能学会、IEEE各会員。

若林 伸和 (正会員)



1965年生。1989年電気通信大学大学院電子情報学専攻修士課程修了。1992年大阪大学大学院工学研究科通信工学専攻博士後期課程単位取得退学。同年、電気通信大学大学院情報システム学研究科助手。1995年静岡大学工学部システム工学科助教授、現在に至る。博士(工学)。画像処理応用計測システム、航海情報データベース、産業応用システム等の開発に関する研究に従事。電気学会、日本航海学会、電子情報通信学会、IEEE各会員。

形状認識部の処理

```

? (system)
Please input cad-data-file number? ==> 1000

(EDGE (SLINE (0 0) (12 0)) (SLINE (12 0) (12 9))
      (SLINE (12 9) (4 6)) (SLINE (4 6) (4 3))
      (SLINE (4 3) (0 3)) (SLINE (0 3) (0 0)))
= 4 pieces
==> ((0 0) (12 0) (12 9) (4 6) (4 3) (0 3)) ← 座標データ

>>>>>> CASE BASED REASONING <<<<<<<
(UNKNOWN (0 0) (12 0) (12 9) (4 6) (4 3) (0 3)) ← 形状認識座標データ

>>>>>> SHAPE APPROXIMATE METHOD 1 <<<<<<<
(UNKNOWN (0 0) (12 0) (12 9) (4 6) (4 3) (0 3))
Approximate axis
==> ((0 0) (12 0) (12 9) (4 6) (0 3))

>>>>>> CASE BASED REASONING <<<<<<<
(UNKNOWN (0 0) (12 0) (12 9) (4 6) (0 3))

>>>>>> SHAPE APPROXIMATE METHOD 2 <<<<<<<
(UNKNOWN (0 0) (12 0) (12 9) (4 6) (0 3))
Approximate axis
==> ((0 0) (12 0) (12 9) (0 9/2))

>>>>>> CASE BASED REASONING <<<<<<<
(DAIKEI (0 0) (12 0) (12 9) (0 9/2)) ← 形状認識された座標データ

>>>>>> SHAPE APPROXIMATE METHOD3 <<<<<<<
No apply

配置計画部の処理
Now System is planning to position the shape:
(DAIKEI (0 0) (12 0) (12 9) (4 6) (4 3) (0 3))

Vertical direction placement start
Lateral direction placement start

Vertical direction placement ==> (0 24)
Lateral direction placement ==> (42 0)
NIL

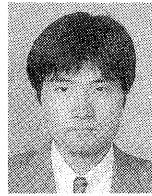
```

図14 プロトタイプシステムの処理過程例

Fig. 14 Example of process on prototype system.

**本多 中二**

1946 年生。1970 年電気通信大学卒業。1972 年同大学大学院修士課程修了。同年同大学助手。現在同大学電子情報学科助教授。工学博士。情報システム、ファジィシステムエキスパートシステムの研究に従事。IEEE, 計測自動制御学会, 日本ファジィ学会各会員。

**田中 功二**

1968 年生。1992 年電気通信大学機械制御工学科卒業。1994 年同大学大学院情報システム学研究科博士前期課程修了。同年, 真如苑企画部情報管理課事務局員, 現在に至る。在学中, CAD/CAM システムの研究に従事。