

マクロ機能による CAD と CAM システムの結合*

長坂 保美*¹, 大滝 英征*²
石川 義雄*² 綿貫 啓一*²

Connection of CAD and CAM using Macrofunction

Yasumi NAGASAKA, Hideyuki OTAKI,
Yoshio ISHIKAWA and Keiichi WATANUKI

The macrofunction in Computer Aided Manufacturing (CAM) systems is executed based on the macrocode which is transformed by a series of operations input by the user. By the use of this function, operations which should be performed by the user can be simplified. Therefore, if the transformation from mechanical drawings to NC code can be changed directly into macrocode, NC machining can be achieved easily. In our study, a function for transforming macrocode from data of a CAD drawing to CAM data is developed. However, drawing data of a CAD system (CAD drawing) does not always coincide with drawing data of a CAM system (CAM drawing), because in a CAM system, restricted conditions, such as machining conditions, must be considered. Consequently, drawing data must be represented by a structure and function which include information about restricted conditions. Thus, the concept of the relationship between "master and servant" for expressing structure and function in an object model definitely is specifically incorporated. As a result, even in the case where structure and function are changed, the knowledge base can be consistently maintained. This paper describes representation of knowledge in the knowledge base and a method for generating a macro code using this knowledge base.

Key Words: Expert System, Artificial Intelligence, CAD, CAM, NC, Macrofunction, Knowledge Base

1. 緒 言

CAD/CAM システムで障害となっているのは、CAD システムの情報 (図面情報) を、CAM システムの情報 (図形情報) として直接的に利用できないことである。その理由は、図面情報 (例えば IGES: Initial Graphics Exchange Specification) では、線要素 (直線や楕円など) に方向性を持っていないのに対し、図形情報 (例えば EXAPT: Extended Subset of Automatically Programmed Tools) では、その線要素の方向を持たせ、工具経路の方向を決定する必要がある。さらに、図形情報には工具選択や、加工条件の設定などの加工情報を含ませる必要もある。そのため、CAM システムに渡された図面情報に対して、ユーザはまず線要素の方向を確認し、その上で加工情報を逐一付加する必要がある。

そこで、設計段階で加工情報を属性として図面情報に付加する研究も多く報告されている。特に、フィーチャモデル (特徴によって名前づけられた対象をモデルの単位として表現) に代表されるように、図面の

形状や加工情報を知識として構成するモデルにより、CAM システムへ橋渡しを行っている。例えば、沖野の TIPS (Technical Information Processing System)⁽¹⁾⁽²⁾ などが代表的なものである。

これらの研究成果を踏まえると、図面情報の線要素を認識し、CAM システム上でフィーチャモデルと同様なモデル表現が構築できれば、図面情報を基にした図形情報が生成できるものと考えられる。これにより、図形情報の生成から NC 加工に至るまでのユーザの操作 (EXAPT システムにおけるパートプログラムに相当) を、システムが図面情報に応じて自動的に実行し得るなら、CAD から NC 加工に至るまでの一貫した自動処理が大いに期待できる。

そのためには、まず図面情報における線要素の認識と、認識の際のルール生成が要求されるが、これについては既報⁽⁴⁾にて述べた。次いで、図面情報を基に、図形情報を生成、管理できるモデルの構築が必要となる。これについては、新たに対象を機能と構造とで表現する対象モデルの概念⁽³⁾を導入した知識表現でフィーチャモデル (知識ベース) の構築を行う。これにより、図形情報を生成する際に、既報⁽⁷⁾で述べた機械加工の加工情報を組み込み、最適な条件を機能と構造とから検討することができる。そして、一貫した自動

* 原稿受付 平成 4 年 8 月 25 日。

^{*1} 埼玉大学大学院 (〒338 浦和市下大久保 255)。^{*2} 埼玉大学工学部。

処理については、この知識ベースを用いて、従来のユーザ操作（図形情報の生成から NC コード生成に至る一連の操作）をコード化（マクロ情報）し、CAM システムのマクロ機能（マクロ情報を基にシステムが動作する機能）により、NC コードを生成することを実現した。これにより、CAD と CAM の結合が容易となり、CAM システムにおける面倒な加工情報の入力操作がなくなり、NC 加工が効果的にかつ容易に行えるようになる。

本稿では、上記モデル構築の知識表現と、マクロ情報の生成の方法、および加工結果について述べている。

2. 知識ベースの知識表現

2.1 本システムの構成と処理過程 図 1 に、本システムの構成と処理過程を示した。CAD システム (CADKEY) と AI ツール [ZES⁽⁵⁾⁽⁶⁾: 著者らの開発による] はワークステーション上で、CAM システム (Smart CAM) はパソコン上で稼働しており、これらはイーサネットを介して互いに情報交換が行われる。また、NC 工作機械 (CAMM-3) は、CAM システムと RS 232 C ケーブルで結合されている。ここで、処理過程を順に追ってみる。

(1) CAD システムでの図面情報 (CAD-drawing) は、CADL (CAD Language) フォーマットで格納されており、IGES に準拠した形式である。また、知識ベースは、2.2 節で述べるような知識表現で格納されている。図面情報は、既報⁽⁴⁾の図面の認識の手法を用いて、この知識表現のインスタンスに格納される

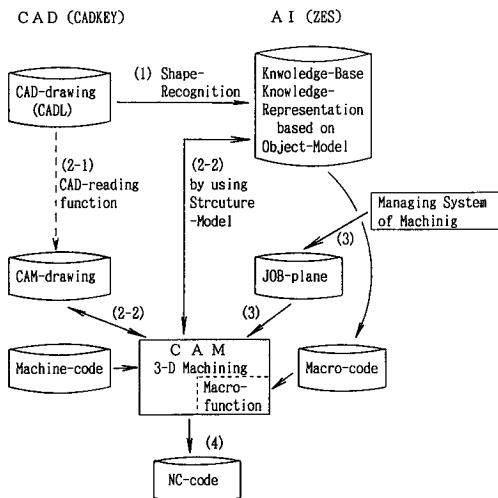


図 1 本システムの構成と処理過程

[図中(1)].

(2) 一般の CAM システムにおける図形情報 (CAM-drawing) は、CAD 読込み機能により直接読込み、これにユーザが加工情報などを付加して生成する [図中(2-1)]. しかし、本システムでは、2.3 節で述べるように、知識ベース内の構造モデル表現を用いて、図面情報に工具経路に関する加工情報を同時に付加して生成する [図中(2-2)].

(3) 工具情報 (JOB-plane) は、既報⁽⁷⁾の機械加工管理システムによって自動的に生成される [図中(3)].

(4) 工具情報、図形情報、機械設定情報 (Machine-code: 工作機械に対応したパラメータを設定) の各ファイルをロードして、NC コード情報 (NC-code) を生成 [図中(4)] する。そして、NC 工作機械にこの情報を転送し、NC 加工を実現する。

マクロ情報 (Macro-code) は、上記一連の流れ [図中の(2-2), (3), (4)] をコード化したものである。本システムは、このマクロ情報を基に、先に述べたユーザ操作を代行する。

2.2 知識ベース内の知識表現 図 2 は、対象モデルの概念⁽⁸⁾を導入した知識表現の枠組み (本システムではフレーム表現の知識の枠組みをユニットと呼ぶ) を示したものである。

図(a)は、対象モデルの概念における視点 (対象の見る位置)のとらえ方を示したものである。例えば、対象 [A 1] は、対象 [A] の一部であり、対象 [A 11] と [A 12] などから構成されている。

図(b)は、対象 [A 1] におけるユニット内部の表現を示したものである。図中の「HasPart, Part-of」は、全体-部分の関係を示したものである。これに対し、図中の「HasStructure, Structure-of」および「HasFunction, Function-of」の関係は、対象モデルの概念の構造と機能を明確に表現するための主従関係を示すものである。「HasStructure, Structure-of」の関係は、対象モデル概念の構造 (2.3 節で述べる対象の構造モデルの表現) の主従関係を示すものである。同様に、「HasFunction, Function-of」の関係は、対象モデル概念の機能 (2.3 節で述べる対象の機能モデルの表現) の主従関係を示すものである。

本システムの知識表現は、上記二つの主従関係「HasStructure, HasFunction」に関する概念により、万一加工条件などによる機能や構造の変更が生じた場合でも、知識ベースを矛盾なく管理する。これについては、2.4 節で述べる。

2.3 構造モデル表現と機能モデル表現 図 3 は、

一例として、燃料噴射ポンプ用カムの構造モデル表現における基本構造テンプレートとルール表現⁽⁴⁾を示したものである。

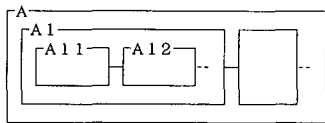
図左は、カムの基本構造テンプレートが、線要素「Base-Arc, Lift-Line, Top-Arc, Side-Arc」から構成されていることを示している。また、図4の機能モデル表現を満たすため、必要に応じて点線表示の円弧が、部分構造テンプレート⁽⁴⁾として用意されている。

図右は、CAD システムの図面情報より、構造テンプレート内の線要素を抽出するためのルール表現である。図例の「Base-Arc」のルール表現は、すでに線要素「Base-Circle」が認識されており、線種が円弧で実線、そして円弧の中心がその図面の中心を示しているとするなら、この線要素を「Base-Arc」とすることを示している。

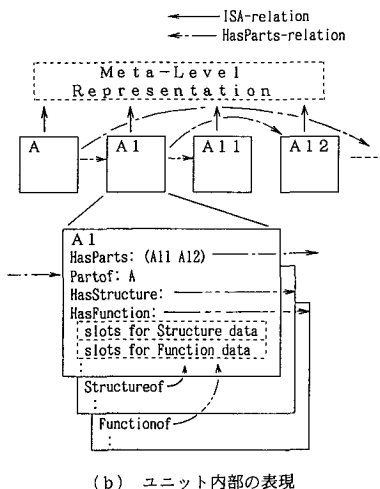
図4は、図3に対応する機能モデル表現における機能要因と運動曲線を示したものである。図中の θ_1 (リフト始め角), θ_2 (変曲角), θ_3 (リフト終り角), リフト距離 (変位), 速度, 加速度などは、モデル表現内のスロットに記述されている。例えば、図中 θ_1 (Lift-Start) のスロットは、以下のように示されている。

Lift-Start :

$$(-180(\text{asin}(/(- \text{Base-Radius} \text{Top-Radius}))) \\ (-(+ \text{Max-Lift} \text{Base-Radius}) \text{Top-Radius}))$$



(a) 対象モデルの概念における視点のとりえ方



(b) ユニット内部の表現

図2 知識ベースの知識表現の枠組み

Radius)))

(if-varied (Base-Radius) (Top-Radius) (Max-Lift))

上の括弧内はスロット値を、下の括弧内は拡張デーモン機能を示すデーモン「if-varied」⁽⁶⁾を示している。このデーモン内の値 (例えば「Base-Radius」) は、スロット値の変数であるとともにスロット名でもある。例えば、スロット「Lift-Start」が参照される場合、最初にこのデーモン機能が起動し、デーモン内の評価 (例えば、スロット「Base-Radius」が参照され、その結果が変数「Base-Radius」に結合) が行われる。そして、スロット値 (上の括弧内) の評価が行われる。これに対し、従来の記述方法では、図中 θ_1 (Lift-Start) に相当するプログラム作成が必要で、これを付加手続きとして呼び出していた。

このように、知識ベース内のモデル表現の記述は、従来の面倒なプログラム作成を必要とせず、従来の表現と同様な表現で記述できる。さらに、上記二つのモデル表現が互いに矛盾なく管理 (2.4 節で述べる) されるので、知識ベースの構築性や可読性に優れていることがわかる。

2.4 モデル表現の管理

図5は、図3と4のモデル表現が互いに矛盾なく管理される過程を示したものである。図例は、スロット「Base-Radius」が、図面情報の入力あるいはユーザーの変更操作によって、知識ベース内がどのように管理されるかを示している。こ

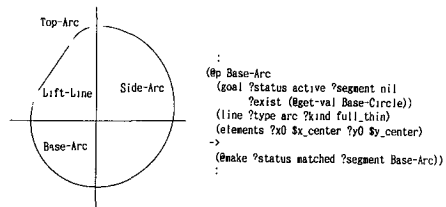
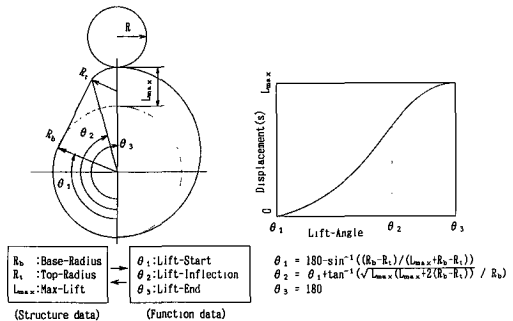


図3 基本構造テンプレートとルール表現



| | | |
|----------------------|------------------------------|---|
| R_b : Base-Radius | θ_1 : Lift-Start | $\theta_1 = 180 \sin^{-1}((R_b - R_1) / (L_{max} + R_b - R_1))$ |
| R : Top-Radius | θ_2 : Lift-Inflection | $\theta_2 = \theta_1 \tan^{-1}(\sqrt{L_{max}^2 - (R_b - R_1)^2} / R_b)$ |
| L_{max} : Max-Lift | θ_3 : Lift-End | $\theta_3 = 180$ |

図4 機能要因と運動曲線

ここで、管理の過程を図中の番号順に追ってみる。

(1) スロット「Base-Radius」の値の変更が起こる。

(2) 主従関係「HasStructure」により、構造モデル表現のユニット「fpcam-###-structure」が参照され、以下の(3)、(4)が行われる。

(3) 構造モデル表現の線要素を示すスロット「segments」を参照し、その値すべてに対して(4)を行う。

(4) アトリビューション⁶⁾「cad」の値で、変更の起きたスロットが使用されているかの検討を行い、使用されている場合は、そのスロット値を変更する。なお、アトリビューションは、スロットの情報を格納するための領域であり、特定の属性を持って(一種のデーモン機能のような)管理を行う。

(5) 主従関係「HasFunction」により、機能モデル表現のユニット「fpcam-###-function」が参照され、以下の(6)、(7)が行われる。

(6) 機能モデル表現の要素を示すスロット「elements」を参照し、その値すべてに対して(7)を行う。

(7) アトリビューション「if-varied」の値で、変更の起きたスロットが使用されているかの検討を行い、使用されている場合は、そのスロットの評価を行い、該当するスロット値にその評価結果を格納する。

このように、本システムの知識ベースは、CAD システムの図面情報の入力に対して、CAM システム上の最適な加工条件の検討に柔軟に対処できる。

3. マクロ機能とコード生成

3.1 CAM システムにおけるマクロ機能の概要

図6は、CAM システム (Smart CAM) のマクロ機能の概念を示したものである。マクロ機能は、ユーザーの一連のキー入力操作を、図7に示したマクロコードを用いてコード化したマクロ情報を基に、システムが代

行するものである。

図7は、マクロコードの種類を示したものである。以下にそのコードの内容を述べる。

「***」形式は、CAM システム内で提供されているコマンドの入力を実現する。例えば、「Files, Save, ...」は、ファイルをセーブする。

「@***」形式は、キーボード上のキー入力を実現する。例えば、「@ cr」はリターン入力を、「@ f5」はファンクションキー f5 と同等の入力を実現する。

「\$***」形式は、コマンド入力に対するデータの入力を実現する。例えば、「\$67」はデータ 67 の入力と同等である。

「#***」形式は、コメント文を表している。

3.2 マクロ情報の生成とコード化 図8は、本システムのマクロ情報の生成機能を示したものである。この生成機能の実行(推論)は、マクロ機能の枠組み(図中の点線枠)の参照を行い、点線枠内の変数「?***」を決定することにより、マクロ情報を生成する。なお、マクロ情報の枠組みは、加工情報の設定からNC加工に至るまでの一連の操作過程をコード化したものである。ここで、マクロ情報の枠組みのコード化について、図中の番号順に追ってみる。

(1) マクロ機能をスタート(@ macstartv 1)させる。

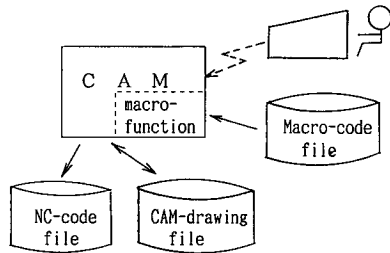


図6 マクロ機能の概念

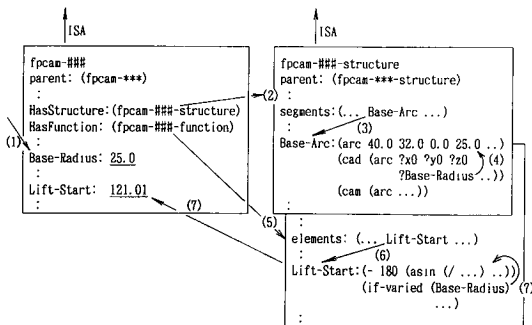


図5 知識ベースの管理過程

```

(1) *** : Input Command
file ... file command
save ... save in a file
code ... create NC-code
:

(2) @*** : Input Key
@macstartv1 ... start macro-function
@cr ... input return-key
@f5 ... input function-key-5
@up ... input control-key-up
:

(3) $*** : Input Data
$67 ... input data "67"
$test1 ... input data "test1"
:

(4) #*** : Input Comment
    
```

図7 マクロコードの種類

(2) 新規 (new) で図面作成を行う。この段階で、
 工具情報 (? tool-file) のロードが行われる。ここで、
 図例の変数「?***」には、スロット「***」の
 値が代入され、(new wax 1 @ cr) を評価、そして工
 具情報「wax 1.jsf」がロードされる。

(3) 図形情報を作成する。スロット「cam-drawing」
 が参照され、以下のリスト式が評価される。

```
(drawing start 0 line 1 line 2 line 3 line 4
start 1 base-arc lift-line ...)
```

ここで、関数「drawing」は、スロット「start 0, line 1,
 line 2...」の順で、アトリビューション「cam」(形状マ
 クロコードと呼ぶ)の評価を行う。例えば、「start 1」
 の場合は、(start_prof \$40.0 @ cr \$7.0 @ cr) と評価
 し、マクロ情報に組込む。図例の「¥second, ¥third」
 は、スロット値の 2 番め、3 番めの値の参照を行うこ
 とを示している。ここで、もし形状マクロコードが未
 定義の場合の処理については、3・3 節で述べる。そし
 て、以下に示すようなマクロ情報が組込まれる。

```
((start_prof $0.0 @ cr $0.0 @ cr)
(line $0.0 @ cr $ 80.0 @ cr)...
(start_prof $40.0 @ cr $7.0 @ cr)
(arc 2. cw $25.0 @ cr $40.0 @ cr accept
```

```
@ cr @ cr @ cr)...)
```

(4) 図形の再描画(@ f 5 full)を行う。

(5) 図形情報(\$ mactest. sh 3)を格納する。この
 段階までが、図形情報を生成するためのマクロ情報で
 ある。ここで、終了することも可能である。

以降は、NC コード生成のためのマクロ情報であ
 る。

(6) データの圧縮(group purge)を行う。

(7) 工具切削経路間の飛越し値(update feature-
 chg clear_z)を設定する。

(8) フライス盤における荒加工(roughing)の処
 理を行う。図例では、外部要素から島形状(island)を
 残した部分を切削(pocket)する。その際、仕上げ代
 (finish), 加工の深さ(depth), パスの深さ(pass-
 depth)などの定義が行われる。なお、これらの値は、
 (2)の工具情報より、材質、工具などによって加工経
 験者の最適な値に設定⁽⁷⁾される。

(9) 工具切削経路のシミュレーション(view)を
 行う。この切削経路に何らかの不都合が生じた場合、
 シミュレーション中に強制終了することも可能であ
 る。

(10) 上記の工具切削経路を基に、NC コードを生
 成し、NC コード情報(mactest. t)に格納する。

このように、2・3 節で述べた機能と構造のモデル表
 現により、最適な図形情報がマクロ情報として格納さ
 れる。さらに、工具情報により、加工情報の最適な値が
 マクロ情報として格納されるので、NC 加工が正確に
 かつ容易に行われる。

3.3 形状マクロコードの生成

3.3.1 図面情報と図形情報 図 9 は、図面情報
 (CAD-drawing)と図形情報(CAM-drawing)の関係
 の例を示したものである。図例の「円弧」の情報は以
 下の形式である。

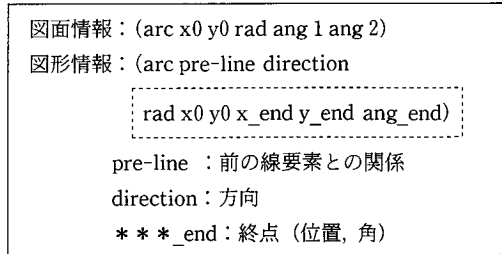
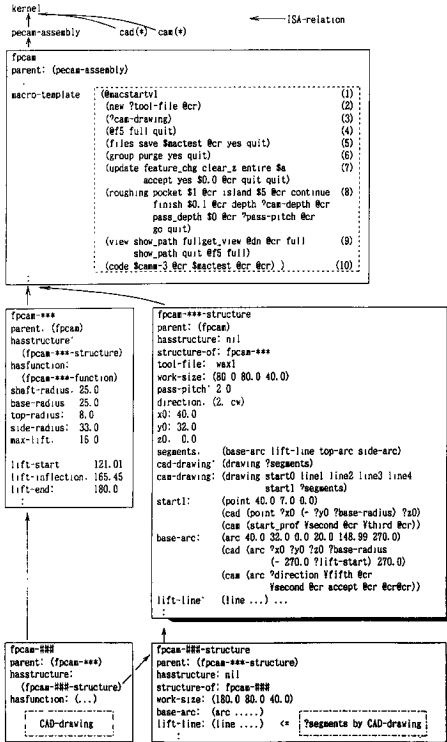


図 8 マクロ情報の生成機能

図形情報の形式は、図面情報の他に、前の線要素と
 の関係 (pre-line: 正接か交差) や方向 (direction: 時
 計方向か反時計方向) などの情報が含まれている。こ
 れは、図形情報の内容が工具経路に直接関係するため
 である。それゆえ、前の線要素との関係と方向が重要

な要素となる。これを管理しているのが、構造モデル表現のユニット (図 8 参照) のスロット「segments」と「direction」である。また、点線枠内の情報は、図 10 で示すように、図面情報を利用して作成することができる。ただし、点線枠内の情報は、前の線要素との関係から自動的に図形情報が決定される場合もあるので、すべての情報を含んでいる必要はない。

図 10 は、上記を勘案した場合の図形情報「円弧」の処理フローである。

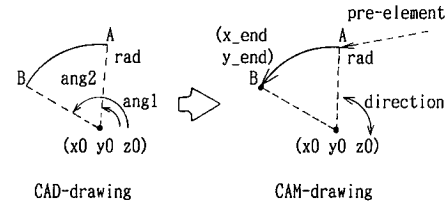


図 9 図面情報と図形情報の関係

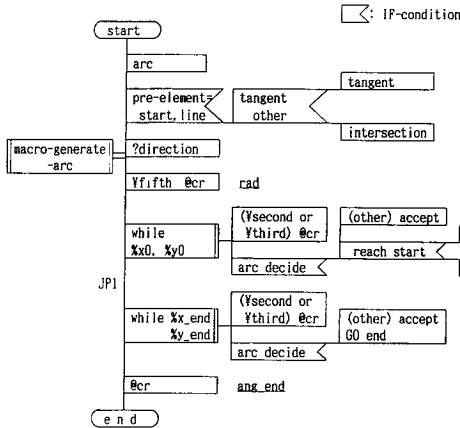


図 10 図形情報「円弧」の処理フロー

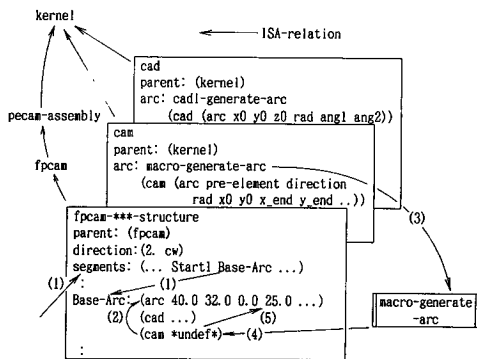


図 11 形状マクロコードの生成過程

ニット内のスロット「direction」の値が組込まれる。また、図中の「%***」は、図面情報の内容を示すもので、ユニット「cad」にそのスロット値の形式が格納されており、該当する「***」の順番が組込まれる。例えば、図中の「%x0」は、上記の図面情報の形式より、「¥second」と評価され組込まれる。

このように、図形情報は、図面情報を基に、構造モデル表現を利用して生成することができる。これにより、CAD システムの図面情報がそのまま CAM システムの情報 (図形情報) として利用できる。

3.3.2 形状マクロコードの生成過程 図 11 は、図形情報「円弧」の場合の形状マクロコードの生成過程を示したものである。図例は、3.2 節の図形情報を作成する段階で、スロット「Base-Arc」の形状マクロコードを参照する場合の過程を示している。この過程を図中の番号順に追ってみる。

(1) スロット「Base-Arc」のアトリビューション「cam」を参照する。ここで、「cam」が未定義の場合は(2)に、定義されている場合は(5)に移る。

(2) スロット「Base-Arc」の値を参照し、線要素が円弧「arc」であることを認識する。

(3) 線要素に応じた形状マクロコードの生成関数 (図例では「macro-generate-arc」) を、付加手続きにより実行する。

(4) 上記の付加手続きを呼び出したスロットのアトリビューション「cam」に、評価結果が格納される。図例では、以下が格納される。

(cam (arc (2. cw) ¥ fifth @ cr ¥ second @ cr accept @ cr @ cr @ cr))

(5) 形状マクロコードの評価が行われ、図形情報を生成するためのマクロ情報として格納される。

このように、形状マクロコードが未定義の場合 (例

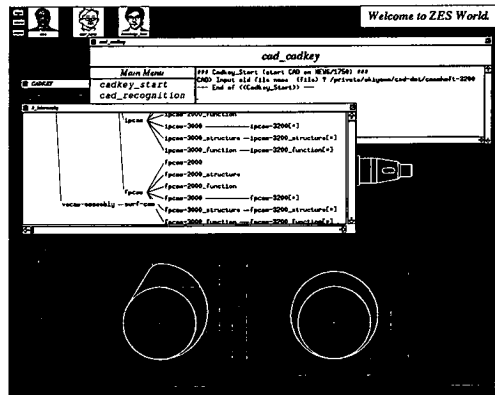


図 12 CAD 図面の認識の実行例

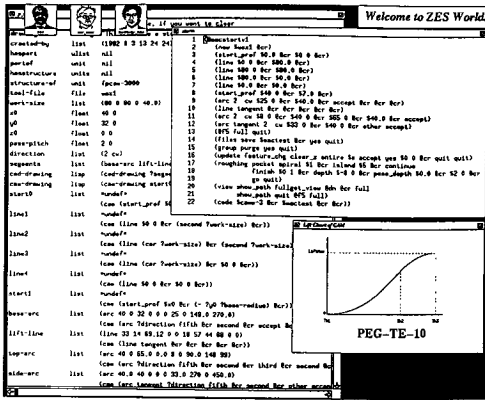


図 13 認識された図面情報による図形情報

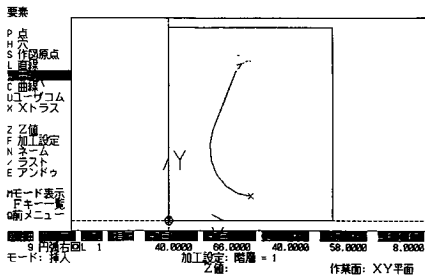


図 14 マクロ機能の実行例 (図形情報の作成)

例えば、CAD システムの類似した図面情報により、新たな線要素のスロットが格納された場合)でも、上記の生成機能により、柔軟に対処することができる。

3.4 実例 図 12 は、AI ツール上から CAD を呼び出し、カム図面と知識ベースの階層構造を示したものである。図中の「cad_recognition」コマンドによって、カム図面の認識された線要素や寸法値が、階層構造上のインスタンスユニット (図中の [*]) に格納される。

図 13 は、図 12 で認識された図面情報による図形情報を示したものである。左部は、構造モデル表現を格納するユニット「fpcam-3 000-structure」の内部表現 (アトリビューション「cam」のみの表示) を示している。また、右上部は生成されたマクロ情報の内容であり、右下部は機能モデル表現の運動曲線 (リフト線図) を示したものである。

図 14 と 15 は、このマクロ情報をパソコン上の CAM システムに転送し、マクロ機能を実行している例を示したものである。図 14 は図形情報の作成段階で、「円弧」形状マクロコードの実行例を示している。また、図 15 は工具切削経路のシミュレーション [図 8

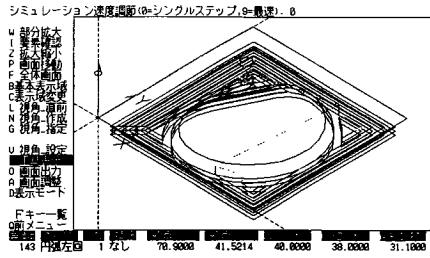


図 15 マクロ機能の実行例 (工具切削経路)

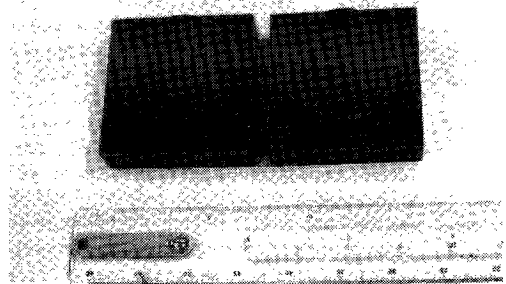


図 16 マクロ機能を用いた NC 加工例

中の(9)] を示しており、図中の点線が工具の移動を表している。

図 16 は、図 12 の CAD システム上の図面情報を基にマクロ情報を生成 (図 13) し、図 14~15 のマクロ機能により、実際に NC 加工を行った結果を示したものである。

4. 結 言

本研究をまとめると、以下のようになる。

(1) 知識ベースの知識表現として、対象を機能と構造とで表現し、これらの関係を矛盾なく管理するための「HasStructure」と「HasFunction」の主従関係を提案した。これにより、設計製造にいたる一連の知識表現の管理が容易になることが期待できる。

(2) CAM システムにおける NC 加工に至るまでの一連の操作入力を、マクロ情報の生成機能により代行するので、製造過程の効率化に役立つ。

(3) 形状マクロコードの生成機能により、CAD システムの図面情報に、柔軟に対応できる。

文 献

- (1) Okino, N., Kakazu, Y. and Kubo, H., TIPS-1; Proc. of

- PROLAMAT '73 (IFIP/IFAC), Budapest, Computer Languages for Numerical Control*, 141-150, (1973), North-Holland Pub. Co..
- (2) Okino, N., Kakazu, Y. and Kubo, H., *TIPS-2, Proc. of PROLAMAT '73 (IFIP) Stirling*, (1976), 385-394, North-Holland Pub. Co..
- (3) 上野, 電子通信学会技法 AI 86-4, (1986), 21-28.
- (4) 長坂・ほか 3 名, 設計工学会誌, 27-12, (1992), 537-542.
- (5) 長坂・ほか 2 名, 機論, 58-547, C (1992), 975.
- (6) 長坂・ほか 2 名, 機論, 58-549, C (1992), 1680.
- (7) 長坂・ほか 3 名, 機論, 58-552, C (1992), 2560.
-