

(19) 日本国特許庁 (J P)

(12) 特 許 公 報 (B 2)

(11) 特許番号

特許第3312141号
(P3312141)

(45) 発行日 平成14年8月5日(2002.8.5)

(24) 登録日 平成14年5月31日(2002.5.31)

(51) Int.Cl. ⁷	識別記号	F I
G 0 6 F 9/44	5 5 4	G 0 6 F 9/44 5 5 4 Z
G 0 5 B 13/02		G 0 5 B 13/02 N

請求項の数18(全 20 頁)

(21) 出願番号 特願平5-339980

(22) 出願日 平成5年12月7日(1993.12.7)

(65) 公開番号 特開平7-160514

(43) 公開日 平成7年6月23日(1995.6.23)

審査請求日 平成11年7月22日(1999.7.22)

(73) 特許権者 000002945

オムロン株式会社

京都市下京区塩小路通堀川東入南不動堂
町801番地

(72) 発明者 久野 敦司

京都府京都市右京区花園土堂町10番地
オムロン株式会社内

(74) 代理人 100080322

弁理士 牛久 健司 (外1名)

審査官 久保 光宏

(56) 参考文献 特開 平4-100152 (J P, A)

特開 平5-46554 (J P, A)

特開 平5-265762 (J P, A)

特開 昭62-276627 (J P, A)

特開 平5-313902 (J P, A)

最終頁に続く

(54) 【発明の名称】 命令ブレーク・ダウン装置および方法ならびに命令解析装置および方法

1

(57) 【特許請求の範囲】

【請求項1】 与えられた上位命令を一または複数の下
位命令に変換し、下位命令を実行できる実行装置に、変
換した下位命令を与える装置であり、

上位命令を実行するために必要な条件を示す知識を記憶
した第1の知識記憶手段、

実行装置の識別符号ごとに、その実行装置によって実現
できる処理の範囲を記憶した第2の知識記憶手段、

上位命令に基づいて、その上位命令の実行のために必要
な条件を上記第1の知識記憶手段から抽出する抽出手
段、

上記抽出手段によって抽出された条件を実現できる処理
の範囲をもつ実行装置の識別符号を、上記第2の知識記
憶手段において検索する検索手段、および上記検索手段
によって検索された識別符号の実行装置に、上記抽出手

2

段によって抽出された条件のうちで該当する実行装置が
実現可能なものの実現を命じる下位命令を与える命令発
行手段を備え、

上記第1の知識記憶手段に記憶された知識には、前件部
と後件部とからなるルールが含まれており、

上記前件部には、実行装置によって実現できる処理の範
囲が記述され、

上記後件部には、上位命令の実行可能範囲が記述され、

上記抽出手段は、上位命令の上記後件部における適合度
を算出し、算出した適合度があらかじめ定められた所定
値以上となる後件部をもつルールを抽出するものであ
り、

上記検索手段は、上記抽出手段によって抽出されたルー
ルの前件部に記述された処理の範囲を実現できる実行装
置の識別符号を検索するものである、

10

命令ブレイク・ダウン装置。

【請求項 2】 上記前件部に実行装置によって実現できる処理の範囲がメンバーシップ関数によって記述されており、

実行装置ごとに、その実行装置が実行できる処理の範囲がメンバーシップ関数で規定されており、

上記検索手段は、上記抽出手段によって抽出されたルールの前件部に記述されたメンバーシップ関数の変数と同じ変数をもつメンバーシップ関数で規定された処理の範囲を実行できる実行装置を検索するものであり、

上記抽出手段によって抽出されたルールの前件部に基づいて実行装置に対する指示値を算出する手段、および上記検索手段によって検索された実行装置の実現可能な処理範囲を規定するメンバーシップ関数に対する上記指示値の適合度を算出し、算出された適合度に基づいて下位命令を与えるべき実行装置を決定する決定手段を備えている、

請求項 1 に記載の命令ブレイク・ダウン装置。

【請求項 3】 上記後件部の上位命令の実行可能範囲がメンバーシップ関数によって記述されており、

上記上位命令が上記後件部メンバーシップ関数の変数値を含むものであり、

上記抽出手段は、上記後件部メンバーシップ関数の上記上位命令に含まれる変数値における適合度が上記あらかじめ定められた所定値以上のルールを抽出するものである、

請求項 2 に記載の命令ブレイク・ダウン装置。

【請求項 4】 上記命令発行手段によって下位命令が与えられた実行装置から、上記前件部メンバーシップ関数の変数値を含む実行結果を受け取る受領手段、および上記受領手段によって受け取られた実行結果に含まれる変数値に基づいて、上記抽出手段によって抽出されたルールをファジィ推論することにより、上記上位命令の実行結果を算出する実行結果算出手段を備えている、

請求項 3 に記載の命令ブレイク・ダウン装置。

【請求項 5】 上記上位命令が他の命令ブレイク・ダウン装置から与えられるものであり、上記実行結果算出手段によって算出された実行結果を、上位命令を与えた他の命令ブレイク・ダウン装置に報告する報告手段を備えている、

請求項 4 に記載の命令ブレイク・ダウン装置。

【請求項 6】 上記実行結果算出手段によって算出された結果を表示する表示手段を備えている、

請求項 4 または 5 に記載の命令ブレイク・ダウン装置。

【請求項 7】 与えられた上位命令を実行装置によって実行させるために、上位命令を実行装置が実行できる処理範囲に分解するための装置であり、

それぞれが、実行装置によって実行できる処理範囲を規定した前件部と実行可能な上位命令の範囲を規定した後件部とからなる複数のルールを記憶したルール記憶手

段、

外部からの上位命令を受付ける命令受領手段、

上記命令受領手段が受け付けた上位命令を実行可能範囲に含む後件部をもつ上記ルールを上記ルール記憶手段から選択する選択手段、および上記選択手段によって選択されたルールの後件部における与えられた上位命令の適合度を算出し、算出された適合度があらかじめ定められた所定値以上である後件部をもつルールの前件部に基づいて、上記上位命令を実行できる実行装置の処理範囲を得る取得手段を備えている、

命令解析装置。

【請求項 8】 上記前件部に実行装置によって実行できる処理範囲がメンバーシップ関数によって記述されており、

実行装置ごとに、その実行装置が実行できる処理範囲がメンバーシップ関数で規定されており、

上記取得手段は、上記選択手段によって選択されたルールの前件部に記述されたメンバーシップ関数の変数と同じ変数をもつメンバーシップ関数で規定された実行装置の処理範囲を得るものである、

請求項 7 に記載の命令解析装置。

【請求項 9】 上記後件部の上位命令の範囲がメンバーシップ関数によって記述されており、

上記上位命令が上記後件部メンバーシップ関数の変数値を含むものであり、

上記選択手段は、上記後件部メンバーシップ関数の上記上位命令に含まれる変数値における適合度が上記あらかじめ定められた所定値以上のルールを選択するものである、

請求項 8 に記載の命令解析装置。

【請求項 10】 与えられた上位命令を一または複数の下位命令に変換し、下位命令を実行できる実行装置に、変換した下位命令を与える方法であり、

上位命令を実行するために必要な条件を示す知識を記憶し、

実行装置の識別符号ごとに、その実行装置によって実現できる処理の範囲を記憶し、

上位命令に基づいて、その上位命令の実行のために必要な条件を上記記憶した条件から抽出し、

上記抽出した条件を実現できる処理の範囲をもつ実行装置の識別符号を、上記記憶した識別符号から検索し、

上記検索した識別符号の実行装置に、上記抽出した条件のうちで該当する実行装置が実現可能なものの実現を命じる下位命令を与え、

上記記憶した知識には、前件部と後件部とからなるルールを含ませ、

上記前件部には、実行装置によって実現できる処理の範囲を記述し、

上記後件部には、上位命令の実行可能範囲を記述し、

上記条件の抽出を、上位命令の上記後件部における適合

度を算出し、算出した適合度があらかじめ定められた所定値以上となる後件部をもつルールを抽出することにより行い、

上記識別符号の検索を、上記抽出したルールの前件部に記述された処理の範囲を実現できる実行装置の識別符号を検索することにより行う、
命令ブレーク・ダウン方法。

【請求項11】 上記前件部に実行装置によって実現できる処理の範囲がメンバーシップ関数によって記述され、

実行装置ごとに、その実行装置が実行できる処理の範囲がメンバーシップ関数で規定され、

上記検索を、上記抽出したルールの前件部に記述されたメンバーシップ関数の変数と同じ変数をもつメンバーシップ関数で規定された処理の範囲を実行できる実行装置を検索することにより行い、

上記抽出したルールの前件部に基いて実行装置に対する指示値を算出し、

上記検索した実行装置の実現可能な処理範囲を規定するメンバーシップ関数に対する上記指示値の適合度を算出し、算出した適合度に基づいて下位命令を与えるべき実行装置を決定する、

請求項10に記載の命令ブレーク・ダウン方法。

【請求項12】 上記後件部の上位命令の実行可能範囲がメンバーシップ関数によって記述されており、
上記上位命令が上記後件部メンバーシップ関数の変数値を含み、

上記条件の抽出を、上記後件部メンバーシップ関数の上記上位命令に含まれる変数値における適合度が上記あらかじめ定められた所定値以上のルールを抽出することにより行う、

請求項11に記載の命令ブレーク・ダウン方法。

【請求項13】 上記下位命令を与えた実行装置から、
上記前件部メンバーシップ関数の変数値を含む実行結果を受け取り、

上記受け取った実行結果に含まれる変数値に基づいて、
上記抽出したルールをファジィ推論することにより、
上記上位命令の実行結果を算出する、

請求項12に記載の命令ブレーク・ダウン方法。

【請求項14】 上記上位命令を命令ブレーク・ダウン方法を
40 実行する他の装置から受け、上記上位命令の実行結果を、
上位命令を与えた命令ブレーク・ダウン方法を実行する装置に報告する、

請求項13に記載の命令ブレーク・ダウン方法。

【請求項15】 上記上位命令の実行結果を表示装置に表示する、

請求項13または14に記載の命令ブレーク・ダウン方法。

【請求項16】 与えられた上位命令を実行装置によって
50 実行させるために、上位命令を実行装置が実行できる
処理範囲に分解するための方法であり、

それぞれが、実行装置によって実行できる処理範囲を規定した前件部と実行可能な上位命令の範囲を規定した後件部とからなる複数のルールを記憶し、

外部からの上位命令を受け、

上記受け付けた上位命令を実行可能範囲を含む後件部をもつ上記ルールを、記憶した複数のルールの中から選択し、

上記選択したルールの後件部における、与えられた上位命令の適合度を算出し、

10 算出した適合度があらかじめ定められた所定値以上である後件部をもつルールの前件部に基いて、上記上位命令を実行できる実行装置の処理範囲を取得する、
命令解析方法。

【請求項17】 上記前件部に実行装置によって実行できる処理範囲がメンバーシップ関数によって記述されており、

実行装置ごとに、その実行装置が実行できる処理範囲がメンバーシップ関数で規定されており、

上記実行装置の処理範囲の取得を、上記選択したルールの前件部に記述されたメンバーシップ関数の変数と同じ変数をもつメンバーシップ関数で規定された実行装置の処理範囲を得ることにより行う、
20 請求項16に記載の命令解析方法。

【請求項18】 上記後件部の上位命令の範囲がメンバーシップ関数によって記述されており、
上記上位命令が上記後件部メンバーシップ関数の変数値を含むものであり、

上記選択を、上記後件部メンバーシップ関数の上記上位命令に含まれる変数値における適合度が上記あらかじめ定められた所定値以上のルールを選択することにより行う、

請求項17に記載の命令解析方法。

【発明の詳細な説明】

【0001】

【技術分野】この発明は、与えられた命令を実行する場合に、その命令を実行するために必要な条件に分解し、その条件を満たすことができる命令を生成し、その命令の実行結果を与えられた命令の実行結果とする命令ブレーク・ダウン装置および方法ならびに命令ブレーク・ダウン処理システムに関する。

【0002】

【背景技術】コンピュータ・システム、データ・ベース・マシン等に代表される情報処理装置では、受け付けることができる命令セット（コマンド・セット）があらかじめ定められて、その装置の内部に記憶されている。したがって、この命令セットに含まれる命令（コマンド）が情報処理装置に入力された場合には、その情報処理装置は、入力された命令を実行することができる。しかし、命令セットに含まれない命令が入力された場合には、その命令は実行できず、命令の入力側に実行不能等

の情報を返すしかなかった。したがって、従来の情報処理装置では、外部から入力される様々な命令に柔軟に対処できないという問題があった。

【0003】

【発明の開示】この発明は、与えられた命令を自らが実行できない場合には、その命令を実行可能な他の命令に分解ないしは置換（ブレイク・ダウン）し、これらの命令を実行できる他の装置に命令を与えることにより、与えられる様々な命令に柔軟に対処できることを目的とする。

【0004】この発明による命令ブレイク・ダウン装置は、与えられた上位命令を一または複数の下位命令に変換し、下位命令を実行できる実行装置に、変換した下位命令を与える装置であり、上位命令を実行するために必要な条件を示す知識を記憶した第1の知識記憶手段、実行装置の識別符号ごとに、その実行装置によって実現できる処理の範囲を記憶した第2の知識記憶手段、上位命令に基づいて、その上位命令の実行のために必要な条件を上記第1の知識記憶手段から抽出する抽出手段、上記抽出手段によって抽出された条件を実現できる処理の範囲をもつ実行装置の識別符号を、上記第2の知識記憶手段において検索する検索手段、および上記検索手段によって検索された識別符号の実行装置に、上記抽出手段によって抽出された条件のうち該当する実行装置が実現可能なものの実現を命じる下位命令を与える命令発行手段を備えている。ここで、上記第1の知識記憶手段に記憶された知識には、前件部と後件部とからなるルールが含まれており、上記前件部には、実行装置によって実現できる処理の範囲が記述され、上記後件部には、上位命令の実行可能範囲が記述され、上記抽出手段は、上位命令の上記後件部における適合度を算出し、算出した適合度があらかじめ定められた所定値以上となる後件部をもつルールを抽出するものであり、上記検索手段は、上記抽出手段によって抽出されたルールの前件部に記述された処理の範囲を実現できる実行装置の識別符号を検索するものである。

【0005】この発明による命令ブレイク・ダウン方法は、与えられた上位命令を一または複数の下位命令に変換し、下位命令を実行できる実行装置に、変換した下位命令を与える方法であり、上位命令を実行するために必要な条件を示す知識を記憶し、実行装置の識別符号ごとに、その実行装置によって実現できる処理の範囲を記憶し、上位命令に基づいて、その上位命令の実行のために必要な条件を上記記憶した条件から抽出し、上記抽出した条件を実現できる処理の範囲をもつ実行装置の識別符号を、上記記憶した識別符号から検索し、上記検索した識別符号の実行装置に、上記抽出した条件のうち該当する実行装置が実現可能なものの実現を命じる下位命令を与えるものである。ここで、上記記憶した知識には、前件部と後件部とからなるルールを含ませ、上記前件部

には、実行装置によって実現できる処理の範囲を記述し、上記後件部には、上位命令の実行可能範囲を記述し、上記条件の抽出を、上位命令の上記後件部における適合度を算出し、算出した適合度があらかじめ定められた所定値以上となる後件部をもつルールを抽出することにより行い、上記識別符号の検索を、上記抽出したルールの前件部に記述された処理の範囲を実現できる実行装置の識別符号を検索することにより行う。

【0006】与えられた上位命令を実行するために必要となる条件を示す知識が、第1の知識記憶手段に記憶される。また、上位命令から変換された下位命令を実行する実行装置の識別符号とその識別符号に対応する実行装置によって実現できる処理の範囲とが、第2の知識記憶手段に記憶される。上位命令が与えられると、その上位命令を実行するために必要な条件が第1の知識記憶手段から抽出される。また、抽出された条件を実現できる処理の範囲をもつ実行装置の識別符号が、第2の知識記憶手段から検索される。検索された識別符号に対応する実行装置に、抽出された条件のうち該当する実行装置が実現可能なものの実現を命じる下位命令が与えられる。ここで、上記記憶した知識には、前件部と後件部とからなるルールを含み、上記前件部には実行装置によって実現できる処理の範囲が記述され、上記後件部には上位命令の実行可能範囲が記述される。そして、上記条件の抽出を、上位命令の上記後件部における適合度を算出し、算出した適合度があらかじめ定められた所定値以上となる後件部をもつルールを抽出することにより行う。また、上記識別符号の検索を、上記抽出したルールの前件部に記述された処理の範囲を実現できる実行装置の識別符号を検索することにより行う。

【0007】この発明によれば、上位命令が与えられた装置が自らその命令を実行することができない場合には、上位命令を、その命令を実行するために必要な条件（命令）にブレイク・ダウン（分解または置換）することができる。そして、ブレイク・ダウンされた条件（命令）を実行することができる他の実行装置に、この条件の実現を命じる下位命令を与えることができる。したがって、命令が与えられた装置自らが実行できない上位命令が与えられても、この上位命令を実行できる他の実行装置に下位命令を実行させることにより上位命令の実行を行うことができる。これにより、与えられた上位命令に柔軟に対処できる。また、抽象的な上位命令を具体的な下位命令に展開することができる。そして、展開された下位命令を、その命令を実行できる適切な実行装置に発することができる。さらに、上記条件の抽出を上記後件部における適合部が所定値以上となる後件部をもつルールを抽出することにより行っているため、上位命令を実行できる可能性の高いルールを抽出することができる。また、抽出されたルールの前件部に記述された処理の範囲を実現できる実行装置の識別符号を検索している

ので、上位命令を実行できる可能性の高い実行装置の識別符号を得ることができる。

【0008】

【0009】好ましくは、上記前件部に実行装置によって実現できる処理の範囲がメンバーシップ関数によって記述され、実行装置ごとに、その実行装置が実行できる処理の範囲がメンバーシップ関数で規定される。そして、上記検索を、上記抽出したルールの前件部に記述されたメンバーシップ関数の変数と同じ変数をもつメンバーシップ関数で規定された処理の範囲を実行できる実行装置を検索することにより行う。また、上記抽出したルールの前件部に基ついて実行装置に対する指示値を算出し、上記検索した実行装置の実現可能な処理範囲を規定するメンバーシップ関数に対する上記指示値の適合度を算出し、算出した適合度に基づいて下位命令を与えるべき実行装置を決定する。これにより、処理範囲がメンバーシップ関数によって記述されるので、実行装置の処理範囲をあいまいな知識として表現することができる。

【0010】この発明の一実施態様においては、上記後件部の上位命令の実行可能範囲がメンバーシップ関数によって記述されており、上記上位命令が上記後件部メンバーシップ関数の変数値を含み、上記条件の抽出を、上記後件部メンバーシップ関数の上記上位命令に含まれる変数値における適合度が上記あらかじめ定められた所定値以上のルールを抽出することにより行う。

【0011】また、上記下位命令を与えた実行装置から、上記前件部メンバーシップ関数の変数値を含む実行結果を受け取り、上記受け取った実行結果に含まれる変数値に基づいて、上記抽出したルールをファジィ推論することにより、上記上位命令の実行結果を算出する。

【0012】この発明の他の実施態様においては、上記上位命令を他の命令ブレイク・ダウン装置から受けることもできるし、上記上位命令を入力手段から受けることもできる。入力手段には、命令ブレイク・ダウン装置に接続されたキーボード、オペレータ等が操作するコンソール等が含まれる。

【0013】また、好ましくは、上記上位命令を他の命令ブレイク・ダウン装置から受け、上記上位命令の実行結果を、上位命令を与えた他の命令ブレイク・ダウン装置に報告する。これにより、上位命令を与えた装置に、上位命令の実行結果を返すことができる。また、他の実施態様においては、上記上位命令の実行結果を表示装置に表示する。これによりオペレータ等は、実行結果を視覚により確認することができる。

【0014】また上記実行装置が他の命令ブレイク・ダウン装置であってもよいし、上記実行装置がセンサおよびアクチュエータを備えたコントローラであってもよい。

【0015】この発明による命令解析装置は、与えられた上位命令を実行装置によって実行させるために、上位

命令を実行装置が実行できる処理範囲に分解するための装置であり、それぞれが、実行装置によって実行できる処理範囲を規定した前件部と実行可能な上位命令の範囲を規定した後件部とからなる複数のルールを記憶したルール記憶手段、外部からの上位命令を受付ける命令受領手段、上記命令受領手段が受付けた上位命令を、実行可能範囲に含む後件部をもつ上記ルールを上記ルール記憶手段から選択する選択手段、および上記選択手段によって選択されたルールの後件部における与えられた上位命令の適合度を算出し、算出された適合度があらかじめ定められた所定値以上である後件部をもつルールの前件部に基ついて、上記上位命令を実行できる実行装置の処理範囲を得る取得手段を備えている。

【0016】この発明による命令解析方法は、与えられた上位命令を実行装置によって実行させるために、上位命令を実行装置が実行できる処理範囲に分解する方法であり、それぞれが、実行装置によって実行できる処理範囲を規定した前件部と実行可能な上位命令の範囲を規定した後件部とからなる複数のルールを記憶し、外部からの上位命令を受付け、上記受付けた上位命令を実行可能範囲に含む後件部をもつ上記ルールを、記憶した複数のルールの中から選択し、上記選択したルールの後件部における、与えられた上位命令の適合度を算出し、算出した適合度があらかじめ定められた所定値以上である後件部をもつルールの前件部に基ついて、上記上位命令を実行できる実行装置の処理範囲を取得するものである。

【0017】それぞれが、実行装置によって実行できる処理範囲を規定した前件部と実行可能な上位命令の範囲を規定した後件部とからなる複数のルールが、ルール記憶手段に記憶される。外部からの上位命令が与えられると、この上位命令を実行可能範囲に含む後件部をもつルールが上記ルール記憶手段から選択される。そして、選択されたルールの後件部における与えられた上位命令の適合度が算出される。算出された適合度があらかじめ定められた所定値以上である後件部をもつルールの前件部に基ついて、上記上位命令を実行できる実行装置の処理範囲が得られる。

【0018】この発明によれば、実行可能な上位命令の範囲を規定したルールの後件部における適合度が所定値以上のルールのみが選択されるので、上位命令を実行できる可能性の高いルールを選択することができる。また、選択されたルールの前件部に基ついて上位命令を実行できる実行装置の処理範囲が得られるので、上位命令を実行できる可能性の高い実行装置を得ることができる。

【0019】好ましくは、上記前件部に実行装置によって実行できる処理範囲がメンバーシップ関数によって記述されており、実行装置ごとに、その実行装置が実行できる処理範囲がメンバーシップ関数で規定されている。

10

20

30

40

50

そして、上記実行装置の処理範囲の取得を、上記選択したルールの前件部に記述されたメンバーシップ関数の変数と同じ変数をもつメンバーシップ関数で規定された実行装置の処理範囲を得ることにより行う。

【0020】さらに好ましくは、上記後件部の上位命令の範囲がメンバーシップ関数によって記述されており、上記上位命令が上記後件部メンバーシップ関数の変数値を含むものであり、上記選択を、上記後件部メンバーシップ関数の上記上位命令に含まれる変数値における適合度が上記あらかじめ定められた所定値以上のルールを選択することにより行う。

【0021】

【実施例の説明】

(1) 命令ブレーク・ダウン処理システムの概要

【0022】図1は、オペレータ等からコンソールを通じて与えられた命令をブレーク・ダウンしながら処理する命令ブレーク・ダウン処理システムの構成を示すブロック図である。命令のブレーク・ダウンとは、与えられた命令を実行するためには、さらに他の複数の命令が実行されなければならない場合に、与えられた命令をこれらの複数の命令に分解ないしは置換することをいう。

【0023】この命令ブレーク・ダウン処理システムは、マルチ・プロセッサ・システムである。この命令ブレーク・ダウン処理システムには、外部バス6、ならびに外部バス6に接続されたコンソール1、n個のプロセッサ21~2nおよび通信装置5が含まれている。プロセッサ21~2nが命令ブレーク・ダウン装置に該当し、命令解析装置を含んでいる。また、コントローラ41~43が命令ブレーク・ダウン装置となり、命令解析装置を含むこともある。

【0024】プロセッサ21には、2つのコントローラ31および32が接続されている。プロセッサ24には、1つのコントローラ33が接続されている。プロセッサによっては、任意の個数のコントローラを接続することもできる。コントローラ31にはセンサ41およびアクチュエータ44が接続されている。コントローラ32にはセンサ42およびアクチュエータ45が、コントローラ33にはセンサ43およびアクチュエータ46がそれぞれ接続されている。コントローラならびにコントローラに接続されたセンサおよびアクチュエータは、n個のプロセッサの全てに接続することもできる。また、プロセッサ21~2nには、表示装置(CRT表示装置、液晶表示装置等)および入力装置(キーボード、ポインティング・デバイス等)の双方または一方が接続される場合がある。

【0025】プロセッサ21~2nは、プロセッサを識別するためのプロセッサId(プロセッサ識別子)「#1」~「#n」を持つ。コンソール1および通信装置5には、識別子として「#0」等のプロセッサ21~2nが持たない例外的な識別子が与えられている。

【0026】この命令ブレーク・ダウン処理システムに

与えられる処理命令は、オペレータ等によってコンソール1を通じて与えられることもあるし、通信装置5を通じて他のシステムから与えられることもある。また、プロセッサ21~2nに入力装置が接続されている場合には、入力装置からプロセッサに直接処理命令が与えられることもある。

【0027】コンソール1、プロセッサ21~2nおよび通信装置5のそれぞれの間では、外部バス6を通じて、後述する命令および報告の送受信が行われる。このとき、上記の識別子によって相手の装置の指定が行われる。

【0028】コンソール1、通信装置5またはプロセッサ21~2nに接続された入力装置から与えられた命令は、後に詳述するが、プロセッサ21~2nによってブレーク・ダウンされた命令に変換され、他のプロセッサもしくはコントローラに与えられるか、または命令を受信したプロセッサ自らによって実行される。実行された命令の結果は、最終的には、命令を与えたコンソール1、通信装置5または入力装置から命令を受け付けたプロセッサに報告される。入力装置から命令を受け付けたプロセッサに命令結果が報告された場合には、その報告はそのプロセッサに接続された表示装置に表示されることもある。

【0029】コントローラ31~33は、プロセッサ21または24から与えられた命令およびセンサからの信号に基づいてアクチュエータを制御する。制御結果は、命令を与えたプロセッサに報告される。

【0030】図2は、この命令ブレーク・ダウン処理システムの処理の概要を示している。処理の一例として、プロセッサ#1(プロセッサ21)に、コンソール1、他のプロセッサ#i(i=2~n)(コンソール1、プロセッサ2i)または通信装置5等から「z=c0にせよ。」という命令81が与えられた場合が示されている。この命令は、「変数zの値を定数c0(クリスプ数)にせよ。」という意味である。定数c0にファジィ数を用いることもできる。

【0031】プロセッサ#1には、以下の知識1および知識2があらかじめ記憶されている。

【0032】〔知識1〕

- 1) 「if x = A1 & y = B1 then z = C1」
- 2) 「if x = A2 & y = B2 then z = C2」
- ...
- m) 「if x = Em & y = Bm then r = Cm」

【0033】〔知識2〕

- 1) 「x = A1, x = A2, ...の範囲は、プロセッサ#3が実行可能。」
- 2) 「y = B1, y = B2, ...の範囲は、プロセッサ#4が実行可能。」
- ...
- 3) 「v = E1, v = Em, ...の範囲は、プロセッサ#2が実行可能。」

【0034】知識1は、ファジィ推論で用いられるル

ルの一つである。知識1の「 $x = A_1 \ \& \ y = B_1$ 」等を前件部といい、「 $z = C_1$ 」等を後件部という。知識1には、上記のように一般に複数のルールが存在する。

【0035】知識1(1)は、「変数 x の値がファジィ数 A_1 (定数)の示す範囲(ファジィ数 A_1 を表すメンバーシップ関数において適合度が0以上の変数域)にあり、かつ変数 y の値がファジィ数 B_1 (定数)の示す範囲(ファジィ数 B_1 を表すメンバーシップ関数において適合度が0以上の変数域)にあるならば、変数 z の値はファジィ数 C_1 (定数)の示す範囲(ファジィ数 C_1 を表すメンバーシップ関数において適合度が0以上の変数域)にある。」ことを意味する。知識1(2)、(m)についても同様である。

【0036】これは逆に考えると「変数 z の値をファジィ数 C_1 に範囲内とするためには、変数 x の値をファジィ数 A_1 の範囲内とし、かつ変数 y の値をファジィ数 B_1 の範囲内としなければならない。」ことを意味する。

【0037】したがって、与えられた命令81「 $z = c_0$ にせよ。」の値 c_0 が後件部のファジィ数 C_1 の範囲内にあるならば、前件部の変数 x の値をファジィ数 A_1 の範囲内とし、かつ変数 y の値をファジィ数 B_1 の範囲内とすれば、 $z = c_0$ または c_0 に近い値にできる可能性があることになる。これは、知識1において、 $z = c_0$ をその範囲に含む(または $z = c_0$ における適合度がより大きいメンバーシップ関数をもつ)ファジィ数を後件部にもつルールを選択すればよいことを意味する。

【0038】すなわち、「変数 x の値をファジィ数 A_1 の範囲内のものとせよ。」という命令(命令1とする)と「変数 y の値をファジィ数 B_1 の範囲内のものとせよ。」という命令(命令2とする)とが実行(達成)されれば、「 $z = c_0$ 」にせよという命令が達成される可能性がある。または $z = c_0$ とはできないにしても、 $z = c_0$ (ファジィ変数 C_1 の示す範囲内の値)は実行(達成)される。

【0039】知識2(1)は、「変数 x の値をファジィ数 A_1, A_2, \dots の範囲内とするのは、プロセッサ#3が実行可能である。」ことを意味する。知識2(2)は、「変数 y の値をファジィ数 B_1, B_2, \dots の範囲内にするのは、プロセッサ#4が実行可能である。」ことを意味する。知識2(n)についても同様である。

【0040】知識2(1)によれば、命令1はプロセッサ#3で実行できることを意味する。また、命令2はプロセッサ#4で実行できることを意味する。したがって、プロセッサ#1は、命令1をプロセッサ#3に、命令2をプロセッサ#4にそれぞれ与えることにより、命令「 $z = c_0$ にせよ。」という命令が実行されることとなる。図2では、この命令1を、ファジィ数 A_1 の示す範囲内にある数値 a_0 (クリスプ数)を用いて「 $y = a_0$ にせよ。」(命令82)という形でプロセッサ#3(プロセッサ23)に与えている。また、命令2を、ファジィ数

B_1 の示す範囲内にある数値 b_0 (クリスプ数)を用いて「 $y = b_0$ にせよ。」(命令83)という形でプロセッサ#4(プロセッサ24)に与えている。

【0041】プロセッサ#3は、与えられた命令1(命令82)を自ら実行し、「 x の現在値は a_1 です。」という報告85をプロセッサ#1に返す。プロセッサ#4は、与えられた命令2(命令83)をさらにブレーク・ダウンし、ブレーク・ダウンされた命令をコントローラ33に与え、実行させる。そして、プロセッサ#4は、コントローラ33からの実行結果の報告を受ける。この報告に基づいて「 y の現在値は b_1 です。」という報告86が作成され、この報告86がプロセッサ#1に送られる。プロセッサ#1では、プロセッサ#3および#4の報告85および86に基づいて、知識1のルール(群)に基づいてファジィ推論を行う。そして、推論結果である「 z の現在値は c_1 です。」という報告84を、命令の発行元(コンソール1またはプロセッサ2i)に返す。

【0042】このように、この命令ブレーク・ダウン処理システムでは、与えられた命令がさらに他の複数の命令に分解または置換され、分解または置換された命令は、この命令を実行できる他のプロセッサまたはコントローラ等に与えられる。与えられた命令をプロセッサ自らまたはコントローラ自らが実行できるならば、そのプロセッサまたはコントローラは、他のプロセッサまたはコントローラ等に命令を与えることなく自ら実行する。このようにして、処理が行われていく。

【0043】以下にこの命令ブレーク・ダウン処理システムの詳細について述べる。

【0044】(2) 命令ブレーク・ダウン処理システムの詳細

【0045】図3は、図1に示されるプロセッサ21(命令ブレーク・ダウン装置)の構成を示すブロック図である。他のプロセッサ2i($i = 2 \sim n$)も、コントローラI/Fの有無(コントローラの接続の有無)の相違を除き、プロセッサ21を同じ構成を有する。

【0046】このプロセッサ21は、CPU70、ROM71、RAM72、ディスクI/F73(ディスク・インタフェース装置)、ディスク装置74~76(磁気記憶装置、光記憶装置等)、外部バスI/F77(外部バス・インタフェース装置)、ならびにコントローラI/F78および79(コントローラ・インタフェース装置)から構成されている。

【0047】CPU70、ROM71、RAM72、ディスクI/F73、外部バスI/F77、ならびにコントローラI/F78および79は内部バスに接続され、内部バスを通じてデータの送受信を行う。

【0048】外部バスI/F77は外部バス6に接続され、この外部バス6を通じて他のプロセッサ、コンソール1または通信装置5との命令および報告の送受信を行う。コントローラI/F78および79は、プロセッサの外

部にあるコントローラ31および32にそれぞれ接続され、これらのコントローラとの間で命令および報告の送受信を行う。

【0049】ROM71には、後に詳述するが、このプロセッサ21の処理プログラムが記憶されている。CPU70は、ROM71に記憶されたプログラムを読みだして、後述する処理を実行する。

【0050】ディスク装置74には、ルール・データおよびメンバーシップ関数データが記憶されている。ルール・データおよびメンバーシップ関数データが、上述した知識1(図2)に該当する。

【0051】図4は、ルール・データの構造を示している。ルール・データは、前件部データと後件部データとからなるルール、各ルールに付けられたルール番号および1または2以上のルールに付けられたルール群番号からなる。前件部データおよび後件部データは、上述した知識1(図2)の前件部および後件部にそれぞれ該当する。

【0052】前件部データは、第1命題、第2命題および第3命題の3つの命題から構成されている。ルールによっては、第1命題から第3命題のすべてを持つものもある。また、第1命題と第2命題しか持たないもの(ルール番号1のルール等)、第1命題しか持たないもの(ルール番号iのルール等)もある。また、この命題の個数を任意として、第4命題、第5命題等を加えることもできる。これらの命題は、ファジィAND演算(MIN演算、激烈積等)によって処理される。以下ではMIN演算による処理について述べる。さらに、各命題は、変数名およびファジィ・ラベルから構成されている。この変数名とファジィ・ラベルとのセットは、後述するメンバーシップ関数データ(図5)に含まれる変数名とファジィ・ラベルとのセットに対応している。例えば、命題の変数名xとファジィ・ラベルA₁とのセットはメンバーシップ関数データの変数名xとファジィ・ラベルA₁とのセットに対応している。

【0053】後件部データは、1つの命題から構成されている。この命題は変数名およびファジィ・ラベルから構成されている。この変数名とファジィ・ラベルとのセットも前件部の命題と同様に、メンバーシップ関数データの変数名とファジィ・ラベルとのセットに対応している。

【0054】ルール番号は、各ルールごとにユニークに付けられている。ここでは、先頭のルールから順番に1~mの連続番号が付けられている。また、後件部データの変数名を同じくする複数のルールは一つのルール群を構成している。このルール群にルール群番号が付けられている。ここでは、後件部データに変数zをもつルール群にはルール群番号1が、後件部データに変数pをもつルール群にはルール群番号2が、後件部データに変数rをもつルール群にはルール群番号Mがそれぞれ付けられ

ている。

【0055】メンバーシップ関数データとは、上記のルール・データにおける変数名とファジィ・ラベルとのセットに対応するメンバーシップ関数を表すデータである。図5は、メンバーシップ関数データの構造を示している。メンバーシップ関数データは、変数名、ファジィ・ラベルおよびメンバーシップ関数のパラメータから構成されている。メンバーシップ関数のパラメータは、さらに4つの値 $x_1 \sim x_4$ からなる。これらの4つの値 $x_1 \sim x_4$ は、メンバーシップ関数上の4つの点の変数値を示している。これらの4つの値によって示される4点の座標は、P₁($x_1, 0$)、P₂($x_2, 1$)、P₃($x_3, 1$)およびP₄($x_4, 0$)と約束されている。この4点を直線で結ぶことによりメンバーシップ関数が作成される。

【0056】 $x_1 < x_2 < x_3 < x_4$ の場合には、図6(a)に示すようにメンバーシップ関数は台形型になる。

$x_1 > x_2$ かつ $x_3 < x_4$ の場合には、図6(b)に示すようにメンバーシップ関数は区間 $[x_3, x_4]$ で傾きをもった直線、区間 $(x_1, x_2]$ でグレードの値が1の直線と約束されている。 $x_1 < x_2$ かつ $x_3 > x_4$ の場合には、図6(c)に示すようにメンバーシップ関数は区間 $[x_1, x_2]$ で傾きをもった直線、区間 $[x_3, x_4)$ でグレードの値が1の直線と約束されている。

$x_1 < x_2 = x_3 < x_4$ の場合には、図6(d)に示すようにメンバーシップ関数は三角形型となる。例えば、変数がx、ファジィ・ラベルがA₁のメンバーシップ関数は、座標(2.0, 0)、(4.0, 1)および(6.0, 0)の3点を結ぶ三角形型のメンバーシップ関数である。

【0057】ディスク装置75には、プロセッサ機能リストが記憶されている。プロセッサ機能リストは、上述した知識2(図2)に該当する。図7は、プロセッサ機能リストの構造を示している。このプロセッサ機能リストは、命令ブレーク・ダウン処理システムに含まれる全プロセッサの個数を示すプロセッサ数(この値はn)、これらのn個のプロセッサI_d、それぞれのプロセッサが処理できる変数名およびメンバーシップ関数のパラメータから構成されている。メンバーシップ関数のパラメータは、上述したメンバーシップ関数データに含まれるパラメータと同じ構成を持つ。

【0058】このプロセッサ機能リストは、どのプロセッサがどの変数をどの程度の適合度で処理できるかを示している。例えば、変数zの値をc₀とする処理について考える。プロセッサ#1が変数zを処理できることが示されている。また、プロセッサ#1のメンバーシップ関数のパラメータで表されるメンバーシップ関数において、z=c₀における適合度が求められる。プロセッサ#1は、この求められた適合度で変数zの値をc₀とする処理ができることとなる。1つのプロセッサが異なる複数の変数を処理できる場合もある。例えば、プロセッ

サ#1が変数zのほかに変数vも処理できる場合等である。この場合には、1つのプロセッサIdに対して、複数の変数名(例えばzとv)およびメンバーシップ関数のパラメータのリストが設けられる。

【0059】ディスク装置76には、命令送信リストが記憶されている。命令送信リストは、プロセッサが与えられた命令を実行するために、他のどのプロセッサにどのような命令を前回送信したかを示すデータである。図8は、命令送信リストの構造を示している。命令送信リストは、受信命令(与えられた命令)の変数名、および受信命令に対して他のプロセッサに与えた第1発行命令、第2発行命令等からなる。各発行命令は、命令の発行先である宛先プロセッサId、変数名および指示値からなる。1つの受信命令に対して、他の複数のプロセッサに命令を発行した場合には、第1発行命令、第2発行命令等の複数の命令の欄が設けられる。例えば、受信命令の変数名がzの欄では、プロセッサ#10に対して「 $x = a$ にせよ。」およびプロセッサ#4に対して「 $y = d$ にせよ。」という2つの命令が発行されたことが示されている。プロセッサ#1が自身に命令を与えることはないの

で、プロセッサ#1のディスク装置76には、プロセッサIdが#1のリストは含まれていない。
【0060】この命令送信リストは命令を送信するごとに更新される。例えば、次回、受信命令に含まれる変数名がpであり、この命令に対して「 $v = w$ にせよ。」をプロセッサ#2に送信すると、受信命令の変数名pの欄の変数名vの指示値がwに更新される。したがって、このリストには、最も近い過去に送信された命令の内容が記憶されていることとなる。また、初期状態において

は、この命令送信リストの各欄には、オペレータ等によって初期値が設定される。
【0061】ディスク装置74~76を1つのディスク装置によって実現できることはいうまでもない。

【0062】RAM72には、関係ルール群エリア、ルール番号/適合度エリア、命令リスト・エリアおよび一時記憶エリアが設けられている。これらの各領域に記憶されるデータについては後に詳述する。

【0063】図9~15は、このプロセッサ21(#1)の処理の流れを示すフローチャートである。このフローチャートによって示される処理は、上述したようにROM71に記憶されたプログラムにしたがって、CPU70が実行する。もちろん他のプロセッサもこれと同様の処理を実行する。

【0064】まず、プロセッサ21は、コンソール1、他のプロセッサ、通信装置5または接続された入力装置からの命令の受信待ちの状態となる(ステップ101)。命令が外部バスI/F77に受信されると、その命令の一部はCPU70によって、その内部レジスタ(またはRAM72の一時記憶エリア)に記憶される(ステップ102)。

【0065】命令の構造は、図16(a)に示されている。

命令は、データの種類、命令が与えられるべき命令の宛先のプロセッサId、命令の発行元のプロセッサId、処理対象となる変数名およびこの変数名で示される変数の達成されるべき値(指示値という)(クリスプ数)からなる。図16(a)では、プロセッサ#5からプロセッサ#1に与えられる「変数zの値を c_0 にせよ。」という命令が示されている。命令の指示値には、ファジィ数を用いることもできる。命令の指示値にファジィ数を用いた場合には、そのファジィ数を表すメンバーシップ関数のパラメータ(図5)等が指示値の欄に置かれるであろう。

【0066】以下では、上記の概要で述べたように、「 $z = c_0$ にせよ(変数zの値を c_0 にせよ)。」という命令が、プロセッサ#5からプロセッサ#1へ与えられ、プロセッサ#1がこの命令を「 $x = a_0$ にせよ。」と「 $y = b_0$ にせよ。」という2つの命令にブレイク・ダウンする例を取り上げて説明を行う。

【0067】与えられた命令のうち、図16(b)に示されるように命令の発行元のプロセッサId、変数名および指示値がCPU70の内部レジスタ(またはRAM72の一時記憶エリア)に記憶される。

【0068】この命令に含まれる変数名と同一の変数名を後件部にもツール(群)が、ディスク装置74のルール・データから読み出される(ステップ103)。読み出されたルールは、RAM72の関係ルール群エリアに記憶される(ステップ104)。図17は、関係ルール群エリアに記憶されたルール群の構造を示している。命令に含まれる変数名はzなので、ルール・データ(図4)のルール群番号1に含まれる、ルール番号1~iのi個のルールが、RAM72の関係ルール群エリアに記憶される。また、ルールの個数iが、この記憶データに付加される。

【0069】次に、関係ルール群エリアに記憶されたルールに含まれる変数名とファジィ・ラベルとのセット(以下では、変数名とファジィ・ラベルとのセットを単にファジィ・ラベルという)に対応するメンバーシップ関数のデータが、ディスク装置74のメンバーシップ関数データから読み出される(ステップ105)。読み出されたメンバーシップ関数データは、ルール・データと同じくRAM72の関係ルール群エリアに記憶される(ステップ106)。

【0070】そして、関係ルール群エリアにある各ルールの後件部データについて、次の処理が行われる。まず、ルール番号1の後件部データのファジィ・ラベル C_i に対応するメンバーシップ関数が、関係ルール群エリアにあるメンバーシップ関数データに基づいて作成される(ステップ107)。そして、作成されたメンバーシップ関数において、変数値(指示値) c_0 における適合度(関数値)が求められる(ステップ108)。求められた適合度は、ROM71等にあらかじめ記憶された第1の基準値と比較される(ステップ109)。求められた適合度

が第1の基準値以上であれば(ステップ109でYES)、そのルール番号と適合度とが組にされてRAM72のルール番号/適合度エリアに記憶される(ステップ110)。求められた適合度が第1の基準値に満たない場合には(ステップ109でNO)、このルール番号と適合度とはルール番号/適合度エリアに記憶されない。

【0071】図18は、ファジィ・ラベル C_1 、 C_2 および C_i のそれぞれに対応するメンバーシップ関数 C_1 、 C_2 および C_i (ファジィ・ラベルと同じ符号をメンバーシップ関数にも用いることとする)を示している。メンバーシップ関数 C_1 については、変数値 c_0 における適合度は d_1 であり第1の基準値以上である。したがって、ルール番号1と適合度 d_1 とはルール番号/適合度エリアに記憶される。メンバーシップ関数 C_2 および C_i では、適合度はそれぞれ d_2 および0であり、いずれも第1の基準値以下である。したがって、これらのルール番号および適合度は記憶されない。

【0072】このように、第1の基準値以上の適合度を有するもののみを選択し記憶するのは、適合度があまりにも小さいルールを選んでも、「 $z = c_0$ にせよ。」という命令を実現できる可能性は小さいので、そのようなルールを採用しないようにするためである。

【0073】これらのステップ107~110の処理が、ルール番号2~ i までの各ルールについても行われる(ステップ111、112)。図19は、これらの処理によってルール番号/適合度エリアに記憶されたデータの構造を示している。ここでは、ルール番号1、3および10が、ステップ110においてルール番号/適合度エリアに記憶されたものとする。ルール番号1について、ステップ108で求められた適合度は d_1 である。ルール番号3および10についての適合度は、それぞれ d_3 および d_{10} である。これは、ファジィ・ラベル C_1 、 C_3 および C_{10} に対応するメンバーシップ関数についての、変数値 c_0 における適合度が第1の基準値以上であったことを示している。

【0074】また、このルール番号/適合度エリアに記憶されたルール番号の個数(ルール数)が計算される。このルール数(この場合は3)が、記憶データに付加されている。ステップ110においてルール番号/適合度エリアに記憶されるべきデータ(ルール番号と適合度)が全くない場合には、このルール数の値は0となる。

【0075】次に、ルール数の値が0かどうか、すなわちルール番号/適合度エリアに記憶されたデータが存在するかどうか判定される(ステップ113)。ルール数は3であり、データは存在するのでステップ114へ処理は進む。データが存在しない場合には(ステップ113でNO)、後述する例外処理(ステップ144~146)となる。

【0076】次に、ルール番号/適合度エリアに記憶された各ルール番号に対応するルールの前件部データの処

理が行われる。まず、ルール番号1に対応するルールの前件部データが、RAM72の関係ルール群エリアから参照される(ステップ114)。そして、参照された前件部データファジィ・ラベル A_1 および B_1 に対応するメンバーシップ関数データが、関係ルール群エリアから参照される(ステップ115)。参照されたメンバーシップ関数データによって作成されるメンバーシップ関数 A_1 および B_1 は、その最大グレード値(=1)が適合度 d_1 (ルール番号1に対応する適合度)になるように変換される(ステップ116)(変換されたメンバーシップ関数を A_{c1} および B_{c1} とする)。

【0077】図20(a)は、ファジィ・ラベル A_1 、 A_3 および A_{10} に対応するメンバーシップ関数 A_1 、 A_3 および A_{10} を示している。ファジィ・ラベル A_1 、 A_3 および A_{10} は、ルール番号/適合度エリアにあるルール番号1、3および10のルールの前件部に含まれるものである。図20(b)は、これらのメンバーシップ関数の最大グレード値(=1)がそれぞれの適合度 d_1 、 d_3 および d_{10} に変換されたメンバーシップ関数 A_{c1} 、 A_{c2} および A_{c10} を示している。

【0078】メンバーシップ関数を、その最大グレード値(=1)が適合度の値になるように変換せずに、メンバーシップ関数を適合度の値で頭切り(トランケーション)してもよい。頭切りによる方法は、処理が簡単である。メンバーシップ関数の最大グレード値を適合度の値にする方法は、メンバーシップ関数の形状が保存されるので、このメンバーシップ関数を用いた演算結果の精度が頭切りによる方法よりも一般に向上するという特徴がある。

【0079】この変換されたメンバーシップ関数のデータは、RAM72の一時記憶エリアに記憶される(ステップ117)。例えば、メンバーシップ関数 A_{c1} は、4つの座標データ(2.0, 0)、(4.0, d_1)、(4.0, d_1)および(6.0, 0)として記憶される。

【0080】これらのステップ114~117の処理が、ルール番号3の前件部データ(ファジィ・ラベル A_3 および B_3)ならびに10(ファジィ・ラベル A_{10} および B_{10})についても行われる(ステップ118、119)。メンバーシップ関数 B_3 および B_{10} がステップ116で変換されたものを、それぞれメンバーシップ関数 B_{c3} および B_{c10} とする。

【0081】一時記憶エリアに記憶された変換後のメンバーシップ関数は、各変数ごとにMAX演算される(重ね合わされる)(ステップ120)。ルール番号1、3および10の各ルールは、その前件部データに変数 x と y のみをもつ。したがって、変数 x については、メンバーシップ関数 A_{c1} 、 A_{c2} および A_{c10} がMAX演算される。変数 y については、メンバーシップ関数 B_{c1} 、 B_{c2} および B_{c10} がMAX演算される。図20(c)は、変数 x についての3つのメンバーシップ関数 A_{c1} 、 A_{c2} および A

C10 がMAX演算された様子を示している。

【0082】次に、MAX演算により得られたメンバーシップ関数について、非ファジィ化(デファジィファイ)処理が行われる(ステップ121)。非ファジィ化の方法には、MAX演算により得られたメンバーシップ関数の面積重心を求めるもの(重心法)や、メンバーシップ関数のグレードが最大となる変数値を求めるもの(最大高さ法)等がある。

【0083】非ファジィ化により得られた値(クリスプ数)は、その変数の代表値とされる(ステップ121)。変数xについての代表値を a_0 (図20(c))、変数yについての代表値 b_0 とそれぞれする。MAX演算されるべき複数のメンバーシップ関数を持たない変数については、1つのメンバーシップ関数に基づいて非ファジィ化処理が行われる。

【0084】次に、ディスク装置75に記憶されたプロセッサ機能リストが参照され、変数xおよびyに対応するリストが読み出される(ステップ122)。変数xについては、プロセッサIdが#3と#n等のリストが読み出される。変数yについては、プロセッサIdが#4等の

リストが読み出される。読み出されたリストは、RAM72の一時記憶エリアに記憶される(ステップ123)。
【0085】読み出された各リストのメンバーシップ関数のパラメータによってメンバーシップ関数が作成される。そして、作成されたメンバーシップ関数において、ステップ121の非ファジィ化処理により得られた代表値における適合度が求められる(ステップ124)。変数xについては、プロセッサ#3、#n等のリストのメンバーシップ関数における代表値 a_0 (変数値)の適合度が求められる。変数yについては、プロセッサ#4等の

リストのメンバーシップ関数における代表値 b_0 (変数値)の適合度が求められる。
【0086】求められた適合度は、ROM71等にあらかじめ記憶されている第2の基準値と比較される(ステップ125)。第2の基準値は、第1の基準値と同じ値であってもよいし、異なる値であってもよい。適合度が第2の基準値以上であれば(ステップ125でYES)、そのリストの変数名、プロセッサIdおよびステップ124で求められた適合度が組にされて、RAM72の一時記憶エリアに記憶される(ステップ126)。適合度が第2の基準

値に満たない場合には(ステップ125でNO)、この変数名、プロセッサIdおよび適合度は記憶されない。
【0087】ステップ124~126の処理が、一時記憶エリアにある全リストについて行われる(ステップ127、128)。この結果、変数xについて、プロセッサ#3とその適合度 h_3 およびプロセッサ#nとその適合度 h_n が、ステップ126で一時記憶エリアに記憶されたものとする。また、変数yについては、プロセッサ#4とその適合度 h_4 が一時記憶エリアに記憶されたものとする。

【0088】全リストについての処理が終了すると(ス

テップ127でYES)、変数xおよびyのそれぞれについて、プロセッサIdおよび適合度が記憶されているかどうかが判定される。まず変数xについてのプロセッサIdと適合度が一時記憶エリアに記憶されているかどうか判定される(ステップ129)。変数xについては、プロセッサ#3と適合度 h_3 およびプロセッサ#nと適合度 h_n が記憶されているので(ステップ129でYES)、ステップ130の処理が行われる。もしプロセッサIdおよび適合度が全く記憶されていない場合には(ステップ129でNO)、後述する例外処理が行われる(ステップ147~149)。

【0089】ステップ130では、変数xについて複数のプロセッサIdおよび適合度が記憶されている場合には、その中で適合度が最大のプロセッサIdが選択される。適合度の高いプロセッサの方が、命令を実行できる(変数xの値を代表値 a_0 にできる)可能性が高いからである。変数xについては、プロセッサ#3および#nが記憶されているので、各プロセッサの適合度 h_3 と h_n とが比較される。ここで、 $h_3 > h_n$ とすると、プロセッサ#3が選択される。

【0090】選択されたプロセッサ#3、変数名xおよび代表値 a_0 は、組(命令リストという)にされて、RAM72の命令リスト・エリアに記憶される(ステップ131)。

【0091】変数yについても、ステップ129~131の処理が行われる(ステップ132、133)。変数yについては、プロセッサ#4、変数名yおよび代表値 b_0 が組(命令リスト)にされて命令リスト・エリアに記憶される。図21は、命令リスト・エリアに記憶された命令リストの構造を示している。命令リストは、宛先プロセッサId(上記のプロセッサId)、変数名および指示値(上記の代表値)から構成されている。また、命令リストには、命令を送るべきプロセッサの個数(対象プロセッサ数、この場合には2)が付けられている。

【0092】命令リスト・エリアに記憶されたデータに基づいて、送信命令が作成される(ステップ134)。作成される命令は、「変数xを a_0 にせよ。」(命令1)および「変数yを b_0 にせよ。」(命令2)である。これらの命令は図16(a)に示される構造をもつ。すなわち、命令1では、データの種別が「命令」、宛先プロセッサIdが「#3」、発行元プロセッサIdが「#1」、変数名が「x」、指示値が「 a_0 」にそれぞれなる。命令2では、データの種別が「命令」、宛先プロセッサIdが「#4」、発行元プロセッサIdが「#1」、変数名が「y」、指示値が「 b_0 」にそれぞれなる。これらの命令は、プロセッサ#3および#4にそれぞれ送信される(ステップ135)。命令の送信後、送信した命令に基づいて、ディスク装置76の命令送信リスト(図8)が更新される(ステップ136)。

【0093】命令を送信したプロセッサは、送信後、あ

らかじめ定められた一定時間、命令を送信したプロセッサからの報告の受信待ち状態となる（ステップ137）。この一定時間は、一般に、命令を与えたプロセッサ（ここではプロセッサ#3および#4）が与えた命令を処理し、処理結果の報告を返すのに十分な時間が選ばれる。

【0094】この一定時間内に、命令を送信したプロセッサ#3および#4から報告が受信されたかどうか判定される（ステップ138）。図16(c)は、報告のデータ構造を示している。報告は、データの種類（報告）、その報告の宛先のプロセッサI d、その報告の発行元のプロセッサI d、報告の対象となる変数名およびこの変数名で示される変数の値（現在値）からなる。図16(c)では、プロセッサ#3からプロセッサ#1に与えられる「変数xの現在値はa₁（クリスプ数）です。」という報告が示されている。報告の現在値に、ファジィ数を用いることもできる。

【0095】ここでは、プロセッサ#1は、プロセッサ#3から「変数xの現在値はa₁です。」という報告およびプロセッサ#4から「変数yの現在値はb₁です。」という報告を受信した場合について考える（報告を受信しない場合については後述する）。

【0096】送信した全てのプロセッサから報告を受信したので（ステップ138でYES）、RAM72の命令リスト・エリアにある命令リスト（図21）の変数xには、現在値a₁が代入される（ステップ139）。また、命令リストの変数yには、現在値b₁が代入される（ステップ139）。次に、これらの変数xおよびyの現在値に基づいて、RAM72の関係ルール群エリアにあるルールがファジィ推論される。ステップ109では、ルール番号1、3および10の3つのルールが選ばれているので、この3つのルールに基づいてファジィ推論が行われる。

【0097】すなわち、ルール番号1、3および10の各ルールの変数xおよびyにa₁およびb₁がそれぞれ代入され、前件部の適合度が求められる。各ルールの後件部メンバーシップ関数の最大グレード値（=1）は、前件部の適合度の値になるように変換される（または頭切りされる）。各ルールの後件部の変換されたメンバーシップ関数（または頭切りされたメンバーシップ関数）がMAX演算される。そして、その面積重心の（または最大の適合度を有する）変数値（変数zの値であり、c₁とする）が求められる（ステップ140）。この求められたzの値c₁は、zの現在値とされる。

【0098】現在値c₁に基づき、図16(c)に示される報告が作成される（ステップ141）。作成された報告は、命令の発行元プロセッサ（プロセッサ#5）に送信される（ステップ142）。発行元のプロセッサI dは、上述したようにCPU70の内部レジスタ（またはRAM72の一時記憶エリア）に記憶されているので、それが参照される。報告が送信された後、処理を終了するかどうか判定される（ステップ143）。例えば、プロセッサ

#1に接続された入力装置によって、終了命令がプロセッサ#1に与えられているならば（ステップ143でYES）、処理は終了する。そうでない場合には（ステップ143でNO）、再び命令の受信待ち状態となる（ステップ101）。

【0099】ステップ113において、RAM72のルール番号/適合度エリアにルール番号と適合度とが記憶されていない場合には、ステップ144～146の処理（例外処理）が行われる。これは、ルールの後件部の適合度が第1の基準値以上であるルールが存在しかつた場合である。この場合には、ディスク装置76に記憶されている命令送信リストが参照される（ステップ144）。コンソールまたは他のプロセッサ等から今回与えられた命令（受信命令）に含まれる変数名と同じ変数名をもつリストが、命令送信リストから参照される（ステップ145）。例えば、与えられた命令に変数名zを含む場合には、図8の命令送信リストにおいて受信命令の変数名zのリストが参照される。

【0100】次に、参照されたリストの第1、第2の発行命令等にしたがって命令が発行される（ステップ146）。図8に示されるリストによれば、プロセッサ#10に「x = aにせよ。」およびプロセッサ#4に「y = dにせよ。」という命令が送信されることとなる。そして、ステップ137以降の処理が行われる。

【0101】また、ステップ126でプロセッサI dおよび適合度が、RAM72の一時記憶エリアに記憶されなかった変数が存在する場合には（ステップ129でNO）、その変数についてはステップ147～149の例外処理が実行される。これは、プロセッサ機能リストにおいて、第2の基準値以上の適合度をもつプロセッサが見つからなかった場合である。この場合にも、ディスク装置76に記憶されている命令送信リストが参照される（ステップ147）。コンソールまたは他のプロセッサ等から今回与えられた命令（受信命令）に含まれる変数名と同じ変数名をもつリストが、命令送信リストから参照される（ステップ148）。例えば、与えられた命令に変数名zを含む場合には、図8に示される命令送信リストにおいて受信命令の変数名zのリストが参照される。

【0102】続いて、命令を送信すべきプロセッサが未定の変数名と同じ変数名をもつ発行命令が参照される。例えば、変数xについてのプロセッサが未定の場合には、変数名zの欄の第1発行命令が参照される。参照された発行命令のデータ（宛先プロセッサI d、変数名、指示値）が、RAM72の命令リスト・エリアに記憶される（ステップ149）。そして、ステップ132から処理が継続される。

【0103】ステップ138において、命令を与えたプロセッサの全てまたは一部から報告が返ってこない場合には（ステップ138でNO）、ステップ150の処理が行われる。すなわち、関係ルール群エリアに記憶されたルール

において、報告が返ってこないプロセッサに与えた命令に含まれる変数名と同じ変数名をもつルールの前件部の適合度が0にされる(ステップ150)。例えば、プロセッサ#3に「x = aにせよ。」という命令を与え、プロセッサ#3から報告が返ってこない場合には、変数名xを含むルールの適合度が0とされる。報告が返ってきた変数については、上述したように報告に含まれる現在値が命令リストの変数に代入される(ステップ139)。そして、ステップ140のファジィ推論が行われる。

【0104】ステップ113でNOの場合には、ステップ140~146の例外処理をする代わりに、第1の基準値をあらかじめ定められた値よりも小さい値に変更して、再度ステップ109の判定を行うこともできる。第1の基準値を小さくしてもなおステップ113でNOの場合には、ステップ144~146の例外処理を行うこともできる。また、ステップ129でNOの場合にもステップ147~149の例外処理をする代わりに、第2の基準値をあらかじめ定められた値よりも小さくして、再度ステップ125の判定を行うこともできる。そして、第2の基準値を小さくしてもなおステップ129でNOの場合には、ステップ147~149の例外処理を行うこともできる。

【0105】ディスク装置75に記憶されたプロセッサ機能リストにコントローラについてのデータ(コントローラId、変数名、メンバーシップ関数のパラメータ)を記憶させることもできる。これにより、プロセッサがコントローラに命令を与えて、コントローラから報告を受けることもできる。

【0106】また、コントローラがプロセッサと同様の構成または簡単なマイクロコンピュータを備え、上記のプロセッサと同様の処理を行うこともできる。これにより、コントローラがアクチュエータに命令(制御命令)を与えて、センサから報告(信号)を受けることもできる。

【0107】コンピュータ・システム内に生成されるタスクが、上述した各プロセッサまたはコントローラの処理を行うこともできる。そして、コンピュータ・システムによって実行されるマルチ・タスクによって、上記の命令ブレイク・ダウン処理システムと同じ処理を実現することもできる。

【図面の簡単な説明】

【図1】命令ブレイク・ダウン処理システムの構成を示すブロック図である。

【図2】命令ブレイク・ダウン処理システムの処理の概要を示す

【図3】命令ブレイク・ダウン処理システムのプロセッサの構成を示すブロックである。

【図4】ルール・データの構造を示す

【図5】メンバーシップ関数データの構造を示す。

【図6】メンバーシップ関数データにより表されるメンバーシップ関数の例を示す。

【図7】プロセッサ機能リストの構造を示す。

【図8】命令送信リストの構造を示す。

【図9】プロセッサの処理の流れを示すフローチャートである。

【図10】プロセッサの処理の流れを示すフローチャートである。

【図11】プロセッサの処理の流れを示すフローチャートである。

【図12】プロセッサの処理の流れを示すフローチャートである。

【図13】プロセッサの処理の流れを示すフローチャートである。

【図14】プロセッサの処理の流れを示すフローチャートである。

【図15】プロセッサの処理の流れを示すフローチャートである。

【図16】(a)はプロセッサ間で通信される命令の構造を、(b)はプロセッサの内部に記憶される命令の構造を、(c)はプロセッサ間で通信される報告の構造をそれぞれ示す。

【図17】関係ルール群エリアに記憶されているルール群の構造を示す。

【図18】メンバーシップ関数の例と、変数値c₀における適合度を示す。

【図19】ルール番号/適合度エリアに記憶されているデータの構造を示す。

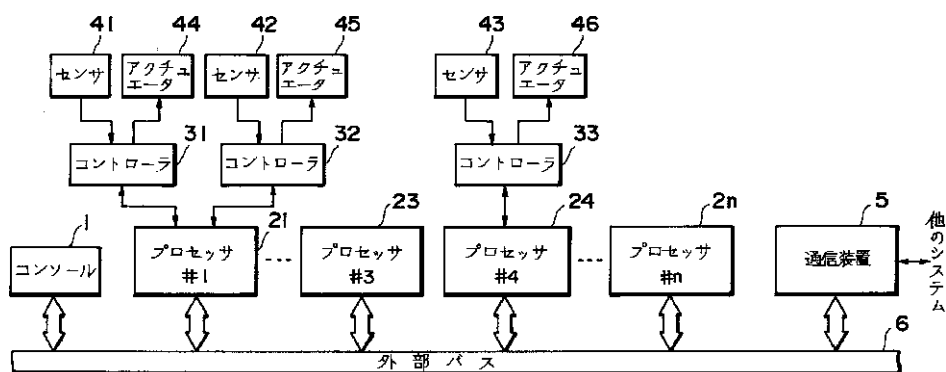
【図20】(a)は3つのメンバーシップ関数の例を、(b)は3つのメンバーシップ関数の最大グレード値が適合度の値になるように変換されたメンバーシップ関数を、(c)は3つのメンバーシップ関数をMAX演算した結果のメンバーシップ関数をそれぞれ示す。

【図21】命令リスト・エリアに記憶されている命令リストの構造を示す。

【符号の説明】

- 1 コンソール
- 21, 23, 24, 2n プロセッサ
- 31, 32, 33 コントローラ
- 41, 42, 43 センサ
- 44, 45, 46 アクチュエータ
- 5 通信装置
- 6 外部バス
- 70 CPU
- 71 ROM
- 72 RAM
- 73 ディスクI/F
- 74, 75, 76 ディスク装置
- 77 外部バスI/F
- 78, 79 コントローラI/F
- 81, 82, 83 命令
- 84, 85, 86 報告

【図1】



【図8】

【図19】

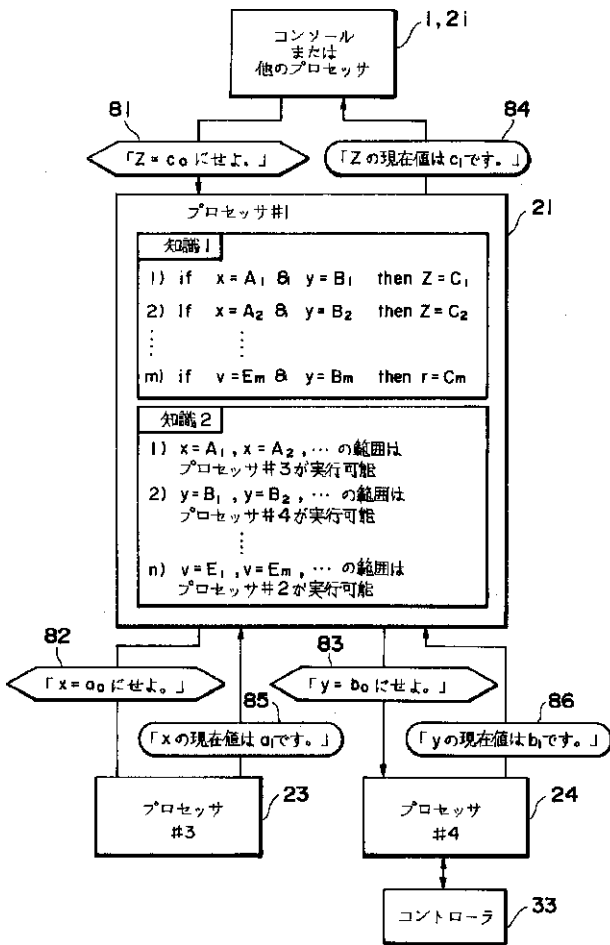
受信命令の変数名	第1発行命令			第2発行命令			ルール番号	適合度
	宛先プロセッサId	変数名	指示値	宛先プロセッサId	変数名	指示値			
z	#10	x	a	#4	y	d	1	d ₁
p	#2	v	e	#10	x	a	3	d ₃
r	#5	w	f	#4	y	b	10	d ₁₀
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮		

ルール数	3
------	---

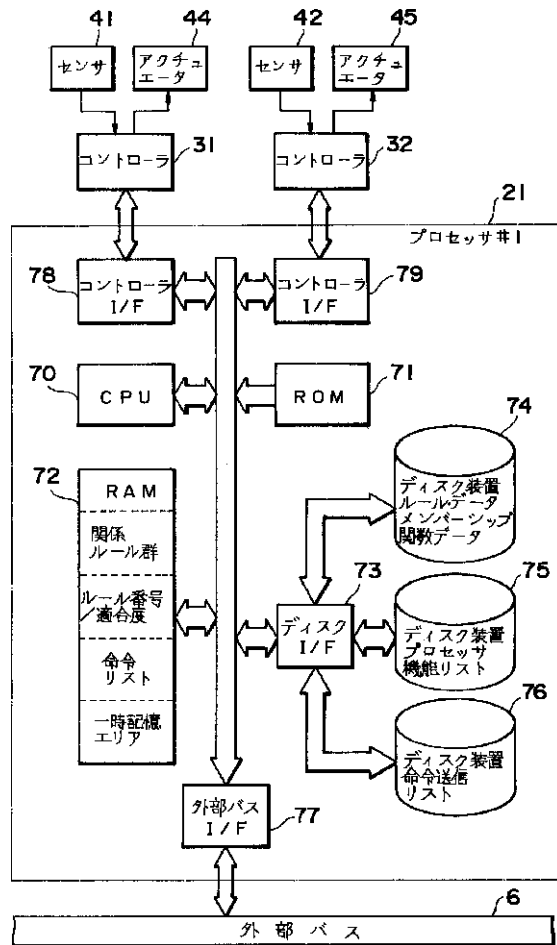
【図4】

ルール群番号	ルール番号	前件部データ						後件部データ	
		第1命題		第2命題		第3命題		変数名	ファジィラベル
		変数名	ファジィラベル	変数名	ファジィラベル	変数名	ファジィラベル		
1	1	x	A ₁	y	B ₁	—	—	z	C ₁
1	2	x	A ₂	y	B ₂	u	D ₂	z	C ₂
1	3	x	A ₃	y	B ₃	—	—	z	C ₃
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	10	x	A ₁₀	y	B ₁₀	—	—	z	C ₁₀
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	i	x	A _i	—	—	—	—	z	C _i
2	i+1	x	A _{i+1}	y	B _{i+1}	v	E _{i+1}	p	C _{i+1}
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
2	j	x	A _j	u	D _j	—	—	p	C _j
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
M	k+1	x	A _{k+1}	w	F _{k+1}	—	—	r	C _{k+1}
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
M	m	v	E _m	y	B _m	—	—	r	C _m

【図2】



【図3】

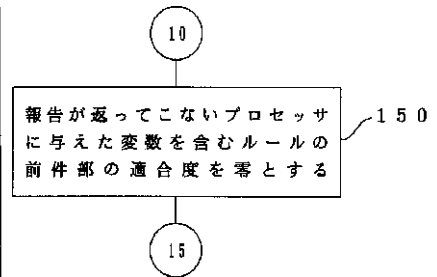


【図7】

プロセッサ Id	変数名	メンバーシップ関数のパラメータ			
		α_{11}	α_{12}	α_{13}	α_{14}
# 1	z	α_{11}	α_{12}	α_{13}	α_{14}
# 2	v	α_{21}	α_{22}	α_{23}	α_{24}
# 3	x	α_{31}	α_{32}	α_{33}	α_{34}
# 4	y	α_{41}	α_{42}	α_{43}	α_{44}
# 5	w	α_{51}	α_{52}	α_{53}	α_{54}
⋮	⋮	⋮	⋮	⋮	⋮
# n	x	α_{n1}	α_{n2}	α_{n3}	α_{n4}

プロセッサ数
n

【図15】



【図21】

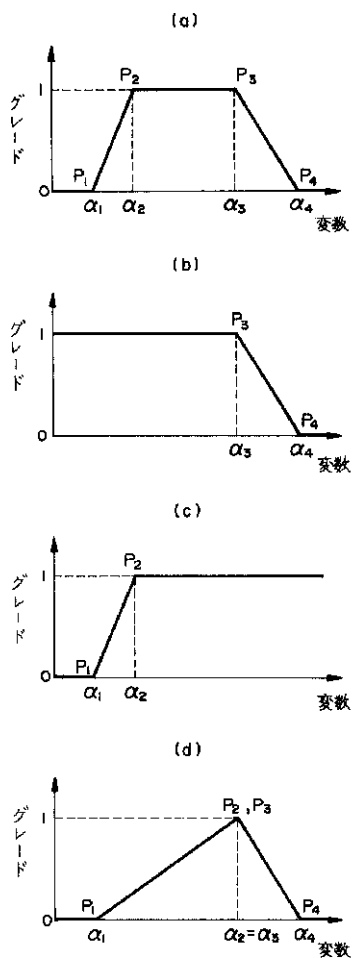
宛先プロセッサId	変数名	指示値
# 3	x	a
# 4	y	b

対象プロセッサ数
2

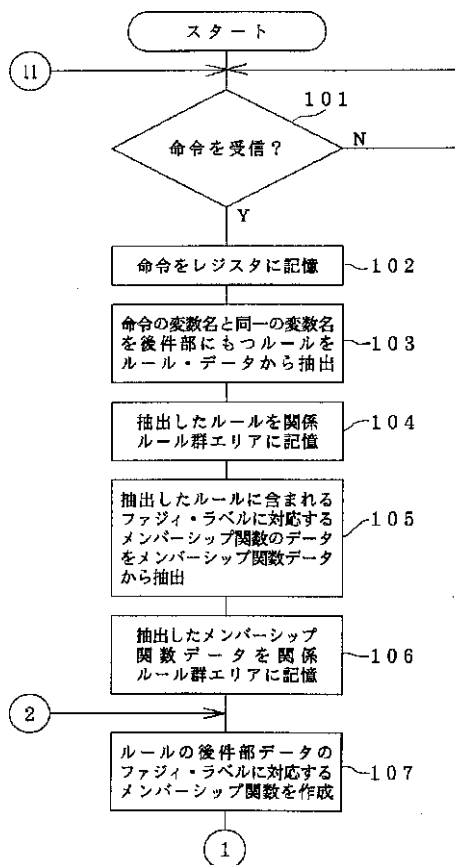
【図5】

変数名	ファジィ・ラベル	メンバーシップ関数のパラメータ			
		α_1	α_2	α_3	α_4
x	A ₁	2.0	4.0	4.0	6.0
x	A ₂	0.0	2.0	2.0	4.0
⋮	⋮	⋮	⋮	⋮	⋮
y	B ₂	0.0	3.5	3.5	5.0
⋮	⋮	⋮	⋮	⋮	⋮
z	C ₁	10.0	15.0	15.0	19.0
z	C ₂	15.0	21.0	23.0	25.0
⋮	⋮	⋮	⋮	⋮	⋮
z	C _i	30.0	33.0	33.0	36.0
u	D ₂	3.5	5.0	5.5	6.0
⋮	⋮	⋮	⋮	⋮	⋮

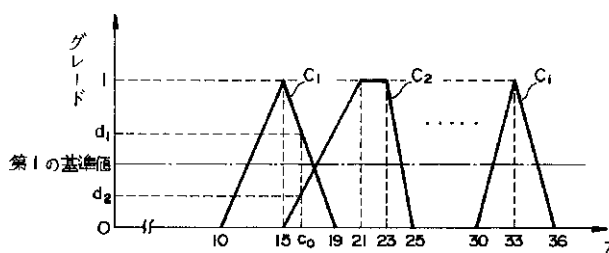
【図6】



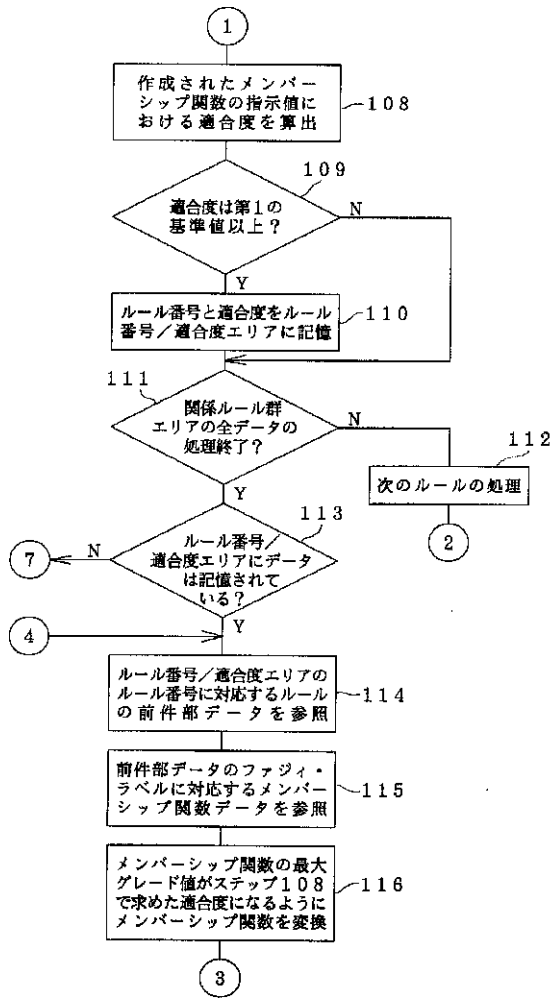
【図9】



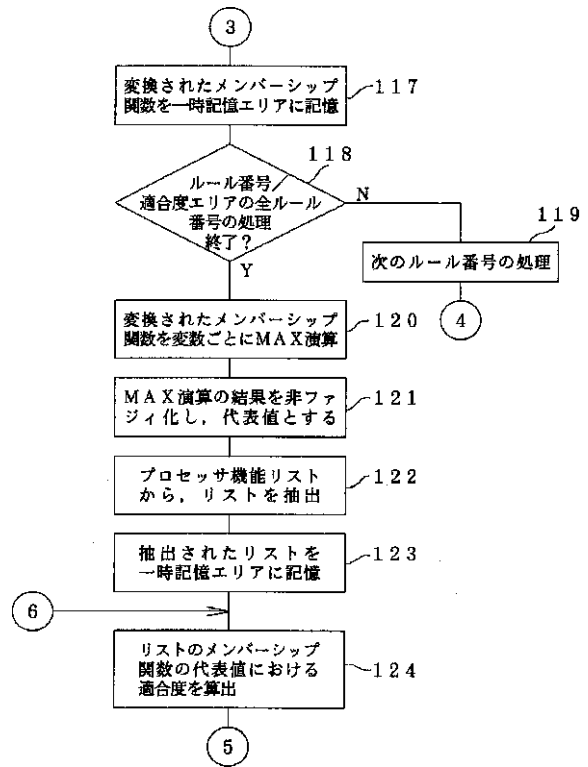
【図18】



【図10】



【図11】

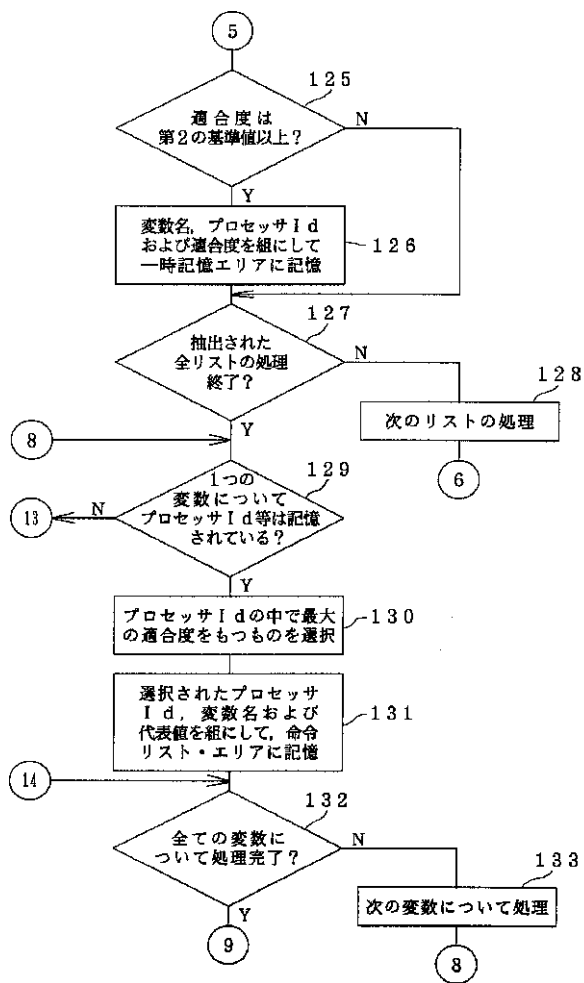


【図17】

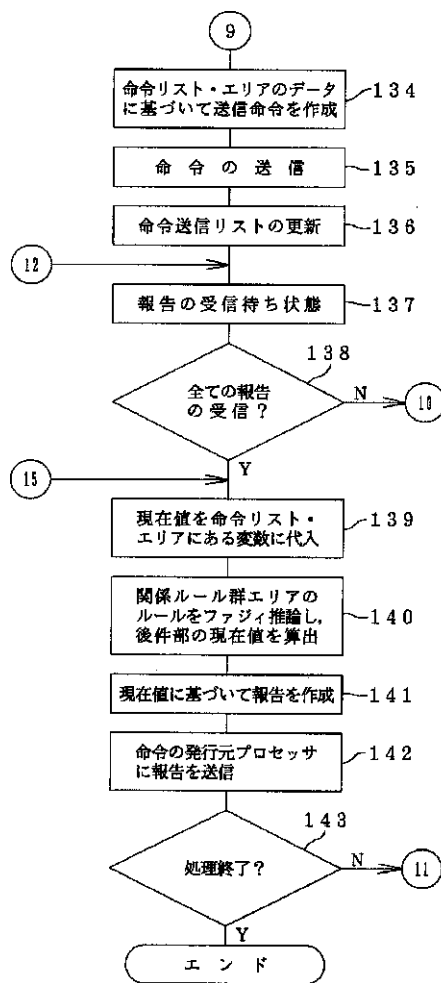
ルール群 番号	ルール 番号	前 件 部 デ ー タ						後 件 部 デ ー タ	
		第 1 命 題		第 2 命 題		第 3 命 題		変数名	ファジィ・ラベル
		変数名	ファジィ・ラベル	変数名	ファジィ・ラベル	変数名	ファジィ・ラベル		
1	1	x	A ₁	y	B ₁	—	—	z	C ₁
1	2	x	A ₂	y	B ₂	u	D ₂	z	C ₂
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	i	x	A _i	—	—	—	—	z	C _i

ルール数
i

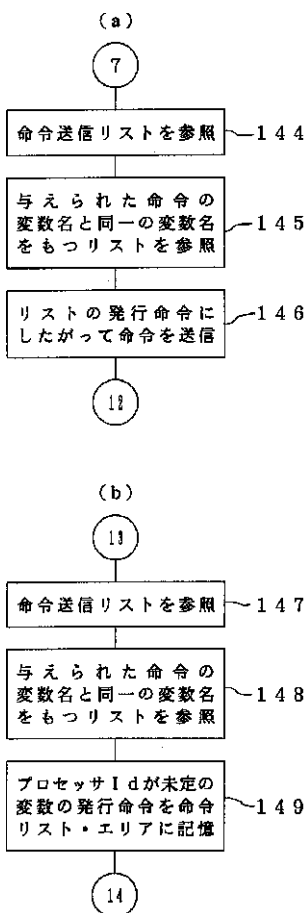
【図12】



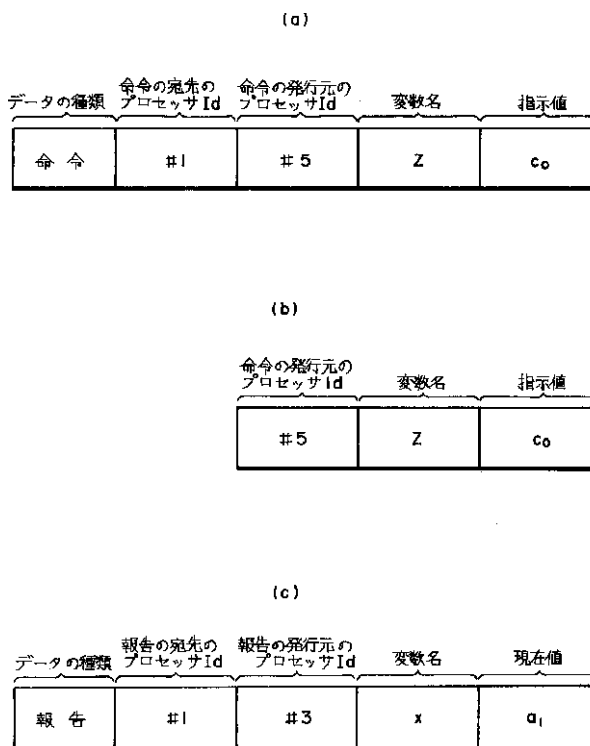
【図13】



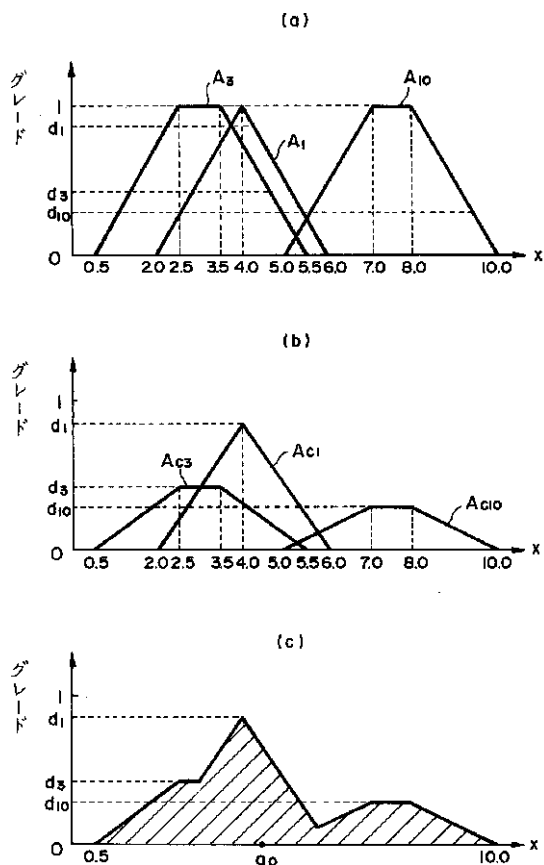
【図14】



【図16】



【図20】



フロントページの続き

(58)調査した分野(Int.Cl.⁷, DB名)

G06F 9/44

G05B 13/02

G05B 15/02

G05B 19/05

G06F 15/16

G06F 9/46

JSTファイル(JOIS)

CSD B (日本国特許庁)