

## 概念管理システム

赤間 清      小笠原 秀美

(北海道大学 文学部)

概念管理システム (CMS : Concept Management System) は, 対象がある属性を持つ / 持たないという形の情報を逐次的に受け取って, それらの情報の間の階層構造を発見し, その構造を仮説として貯え, 逐次的に更新し, それについての質問に答える帰納的学習システムである. 本論文では, 実現したCMSの概要について述べる. そのようなシステムを実現するために必要な知識構造として, 概念階層 (CH) と呼ばれる構造を準備し, その性質を調べる. CHは包含関係により階層的に配置された集合族を表現する枠組みで, 直観的には, そのグラフの頂点に「概念」が対応している. また, CMSによるCHの更新アルゴリズムを検討する. 最後に, 帰納的学習システムLS/1の知的サブシステムとしてCMSが動作し得ることを確認する.

Concept Management System

Kiyoshi AKAMA and Hidemi OGASAWARA

(Faculty of Letters, Hokkaido University, Sapporo-shi, 060, Japan)

Concept management system (CMS) is an inductive learning system that receives successively positive and negative examples of unary relations (stating that a certain object has a certain property), generates the hierarchical structure and answers the query on the relations. This paper explains the outline of the implemented CMS. Its main knowledge representation system is called concept hierarchy (CH). We discuss the structure of CH and the method to represent "concepts" as the extension. We also describe the algorithm to improve CH when the new input information is given. CMS can work as the intelligent subsystem of the learning system LS/1.

## 1. まえがき

概念などのなす継承階層は、知識表現の中で特に重要な役割を果たす。しかしながら、それらの階層構造がどのようにして獲得されるのかについては、ほとんどわかっていない。本論文では、それを解明する糸口として、逐次的に与えられる1項関係の情報から、階層構造を逐次的に構成する帰納的学習システムについて考察する。

その帰納的学習システムは概念管理システム(CMS)と呼ばれる。それは、概念階層(CH)と呼ばれる階層的な構造体を仮説として作成する。概念には外延として集合が対応する。従って、複数個の概念が与えられると、集合の包含関係により階層構造が導入できる。CHはそれを表現している。しかし、学習の観点から見る時、階層構造についてこれとは別の把握が必要である。それは階層構造が概念の獲得を助けるという側面である。これら2つの見方を図式で書けば、それぞれ、A: [概念→階層構造]、B: [階層構造→概念]となり、方向が互いに逆の関係になる。学習過程を抜く場合には、どの時点においても各概念の完全な外延が与えられることはないので、Aだけの見方になることはありえない。CHは、Bの見方をもとにして作られた知識表現である。

## 2. CMSの概要

### 2.1 CMSの設定

概念管理システム(CMS: Concept Management System)は、対象がある属性を持つ/持たないという形の情報を扱う帰納的学習システムである。CMSは以下に述べるような動作をする。実際の入出力は記号の組で為されるが、分かりやすさのため以下では入出力を言葉で表現している。

- (1) CMSに対して情報が逐次的に与えられる。
- (2) 情報は次の形をしている。  
「対象tが属性zを持つ」 例: 象は歩く  
「対象tが属性zを持たない」 例: 馬は飛ばない
- (3) CMSは、それらの情報を総合して、逐次的に種々の「概念」を獲得する。
- (4) CMSは次の形の質問を受け付ける。  
「対象tは属性zを持つか？」 例: 狼は飛ぶか?  
「対象tの属性には何があるか？」 例: 羊の属性は?  
「属性zを持つ対象には何があるか？」 例: 飛ぶのは何か?
- (5) CMSは、獲得した「概念」を利用して質問に答える。  
\*yes/no-質問には、「はい」/「いいえ」で答える。  
\*what-質問には、要求のあるたびに条件を満たす例を答える。  
もう条件を満たす例が残っていなければ、「もうありません」と答える。

### 2.2 学習過程の例

前節の(3)においてCMSの作る主要な知識は、概念階層(CH)と呼ばれる構造体である。CHは「ループのないグラフ」すなわち「森」を中心とした知識表現で、その頂点が概念に対応している。

CMSの学習過程(概念獲得過程)を、例によって説明する。CMSが、ある時点までに、ヒロコから彼女の身のまわりの愛称のついた動物に関して、以下の様な情報を受け取ったと仮定する。

「シロとタロは犬である」  
「シロは動物である」  
「エンジェルは魚であるが犬ではない」  
「エンジェルは動物である」  
「シロはほ乳類であるがエンジェルはほ乳類ではない」  
「ペルシャは犬ではないがほ乳類である」  
「ニョロはほ乳類でもないし犬でもない」  
「シロは私のペットである」  
「エンジェルとペルシャは私のペットではない」

これらの情報は、前節の(2)の形で逐次的に与えられる。CMSは情報を受け取る毎にCHを更新する。以上の情報をこの順に受け取った場合、CMSは図1-1)に概略が示されるようなCHを作り上げる。この時点でCMSは、

「タロは動物ですか？」

の質問に対して、その事実を教えられていないにもかかわらず、「はい」と(正しく)答える。これはCMSの概念階層CHが、与えられた情報の単なる寄せ集めではなく、CMSが推測して作り上げた仮説的知識だからである。またこの後、

「ニョロは私のペットである」

という追加情報を受け取ると、CMSはCHを図1-2)の様に変える。この場合、「私のペット」という属性は、すでに与えられた負の情報のために、「動物」を根とする現在のCHの木構造にうまく位置付けることができない。CMSは、新しい頂点や辺をCHに追加してその情報をCHに追加している。

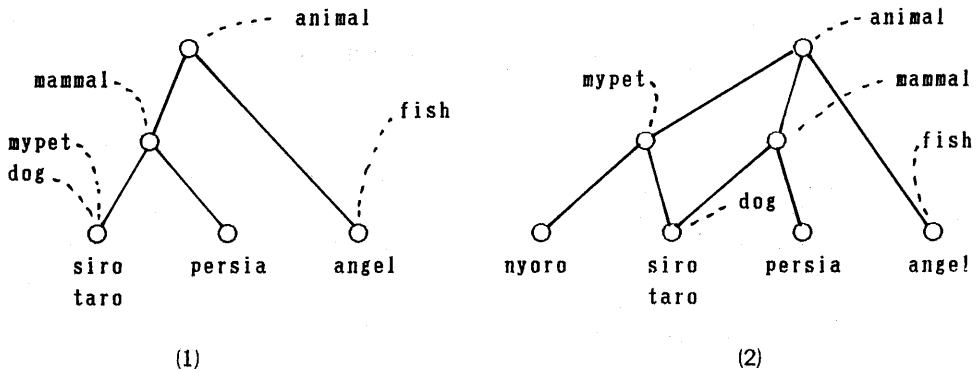


図1 学習によるCHの変化の例

### 3. 概念管理システムの設計問題

上記のようなCMSの設定を改めて形式的に述べ、CMSを設計するために必要な(最小限の)条件を明確にする。

互いに排反な2つの集合：属性集合乙と対象集合丁の存在をあらかじめ仮定する。乙と丁の元をそれぞれ、属性、対象と呼ぶ。CMSへの情報は、符号  $p/n$  と属性  $z$  と対象  $t$  からつくられる3項組

$(p, z, t)$  または  $(n, z, t)$

である。前者は、「対象  $t$  が属性  $z$  を持つ」に、後者は「対象  $t$  が属性  $z$  を持たない」に対応し、それぞれ、正の例、負の例と呼ばれる。

このような情報を次々に与えられて、CMSは逐次的にその知識を更新する。その知識は、少なくとも、 $\mathbb{Z} \times \mathbb{T}$ の部分集合を表現する必要がある。その部分集合をSとするとき、yes/no-質問：「対象tは属性zを持つか」は、「 $(z, t) \in S$ であるか」という問題と等価である。また、what-質問「属性zを持つ対象には何があるか」は、集合 $\{t \mid (z, t) \in S\}$ を見つけて、その元を次々に答える問題に帰着される。

任意の時点において、それまでに得た正負の例から、

$$P = \{(z, t) \mid (p \quad z \quad t) \text{ は、それまでに得た正の例}\}$$

$$N = \{(z, t) \mid (n \quad z \quad t) \text{ は、それまでに得た負の例}\}$$

と定義する。このときSは、条件

$$P \subset S \subset N^c$$

☆

を満たさねばならない。ここでcは $\mathbb{Z} \times \mathbb{T}$ に関する補集合を求める演算子である。

従って、CMSの設計では、少なくとも、次のような問題を解く必要がある。

- (1) CMSの知識表現はどんな方法が適当か。
- (2) ☆を満たしながら知識表現を更新するアルゴリズムはどう作ればよいか。

以下では、我々の作成したCMSについて、上記の2点を検討する。

## 4. 概念階層

### 4.1 概念階層(CH)の定義

CMSの作る主要な知識は、概念階層(CH)と呼ばれる構造である。概念階層CHは、以下の条件(1), (2), (3), (4), (5), (6)を満たす次の5項組である。

$$\langle G, Z, T, f, g \rangle$$

- (1) Gは、頂点(vertex)集合Vと辺(arc)集合Aからなる(有向)グラフ $\langle V, A \rangle$ である。定義より、AはV上での1つの辺を通しての到達可能関係と見ることができる。それを拡張して、V上でのm個の辺を通しての到達可能関係を、Aのm乗すなわち $A^m$ で定義する。このときAの0乗 $A^0$ はV上の恒等関係となる。また、Vの任意の元vに対して、vの子、祖先の集合、子孫の集合を、それぞれ、 $child(v)$ ,  $des(v)$ ,  $anc(v)$ で表わす。ただし、vの子孫と祖先はv自身も含むとする。
- (2) グラフGは次のG1からG5の性質を満足する。
  - G1: ループが存在しない。  
 $\neg \exists v, \exists m > 0 : (v, v) \in A^m$
  - G2: 推移的にもとめられる辺は存在しない。  
 $\exists m > 1 : (v_i, v_j) \in A^m \rightarrow (v_i, v_j) \in A$
  - G3: 2つの頂点に共通の子孫の空でない集合には、対応する頂点が存在する。  
 $des(v_i) \cap des(v_j) \neq \emptyset \rightarrow \exists v \in V : des(v) = des(v_i) \cap des(v_j)$
  - G4: 任意の頂点の子の数は1ではない。  
 $|child(v)| \neq 1$
  - G5: グラフ全体の根が存在する。  
 $\exists v \in V : des(v) = V$
- (3) Zは属性の集合 $\mathbb{Z}$ の部分集合である。
- (4) Tは対象の集合 $\mathbb{T}$ の部分集合である。
- (5) fはTからFの葉Lへの全射である。
- (6) gはZからVへの写像である。

## 4. 2 概念階層の表現する関係, 集合

概念階層  $CH = \langle G, Z, T, f, g \rangle$  があるとする。そのとき  $CH$  は,

$$S = \{ (z, t) \mid g(z) \text{ は } h(t) \text{ の祖先である} \}$$

によって  $Z \times T$  の部分集合を 1 つ与える。これが  $CH$  の決める  $Z \times T$  の部分集合である。 $G = \langle V, A \rangle$  の ( $V$  の) 各頂点  $v$  に、空でない集合を対応させる写像  $F: V \rightarrow P(T)$  を次の式で定義する。

$$F(v) = f^{-1}(v) \quad \dots \quad v \text{ が葉のとき}$$

$$F(v) = \cup F(v') \quad (v' \in \text{child}(v)) \quad \dots \quad v \text{ が子を持つとき}$$

$F$  は次のような単調性を持つ。

$$v_1 \text{ が } v_2 \text{ の子孫} \rightarrow F(v_1) \subset F(v_2)$$

また  $F$  により、 $Z$  に属する任意の属性  $z$  に対して、対象の空でない集合  $F(g(z))$  が対応する。これは属性  $z$  を持つ対象の集合である。

$S$  は、 $CH$  を外から見たときの「意味」である。つまり  $CH$  は  $Z \times T$  の 1 つの部分集合  $S$  とみなされる。それに対して、写像  $F$  は、 $CH$  を内から見たときの様子を表わす。 $CH$  の内部には、 $F(v)$  で示されるいろいろな集合が存在する。

## 4. 3 概念階層 $CH$ と集合族 $\Sigma$

概念階層  $CH$  がどのような集合を表現できるかを考えるために、 $\Sigma = F(V)$  とする。 $\Sigma$  は次の性質  $S1 - S4$  を持つ。逆に  $S1 - S4$  の性質を持つ集合族  $\Sigma$  は、適当な  $CH$  によって  $\Sigma = F(V)$  と書ける。証明は省略する。

$S1$ :  $\Sigma$  の元  $S$  は  $T$  の空でない部分集合である。

$$S \in \Sigma \rightarrow \phi \neq S \subset T$$

$S2$ :  $\Sigma$  の全ての要素は  $\Sigma$  のなかのいくつかの極小な要素の和として表される。

$$S \in \Sigma \rightarrow S = S_1 \cup S_2 \cup \dots \cup S_n \quad \& \quad S_i \quad (i=1, 2, \dots, n) \text{ は } \Sigma \text{ で極小}$$

$S3$ :  $\Sigma$  のある 2 つの集合が共通する要素を持つならその積集合も  $\Sigma$  に含まれる。

$$S_i, S_j \in \Sigma \quad \& \quad S_i \cap S_j \neq \phi \rightarrow S_i \cap S_j \in \Sigma$$

$S4$ :  $\Sigma$  のすべての要素の和集合も  $\Sigma$  の要素である。

$$\cup S_i \in \Sigma$$

これらの集合の関係を  $CH$  はグラフ  $G$  で簡潔に表現している。 $G1$  の条件は、集合の包含関係を表す。 $G2$  はグラフの表現を簡潔にしている。 $G3$  は、ある集合とある集合の積集合をすばやく発見することを可能にする。 $G4$  は同じ集合に対応する頂点を一意にする。 $G5$  は根で全体をまとめ、グラフ  $G$  における探索を容易にする。

## 5. 概念管理システム CMS のアルゴリズム

### 5. 1 $CH$ の操作

CMS は  $CH$  を更新するために行なう操作の例を挙げる。括弧内は関係する変数であり、下でそれを用いて説明する。

基本操作例

- a1 頂点の親/子を追加/削除する。(親頂点  $p$ , 子頂点  $c$ )
- a2 属性に対応する頂点を設定/変更する。(属性  $z$ , 頂点  $v$ )
- a3 対象に対応する葉を設定/変更する。(対象  $t$ , 葉  $v$ )

### 複合操作例

#### (1) 頂点(と辺)の生成

- b1 新しい葉を生成する。(親頂点  $p$ , 対象集合  $\tau$ )
- b2 既存の葉の下に2つの葉を生成する。(葉  $v$ , 対象集合  $\tau$ )
- b3 頂点間に新しい頂点を生成する。(親頂点  $p$ , 子頂点  $c$ )

#### (2) 辺, 頂点の削除

- b4 direct arcを削除する。(親頂点  $p$ , 子頂点  $c$ )
- b5 子が1つである頂点を削除する。(頂点  $v$ )

$a1$ は,  $p$ の子リストに $c$ を,  $c$ の親リストに $p$ を追加し, またそれらから削除する。 $a2$ は $g(z)$ の設定/変更である。 $a3$ は $f(t)$ の設定/変更である。 $b1$ は $p$ の子として新しい葉 $v$ を生成し, 任意の $\tau$ の元 $t$ に対して $f(t) = v$ とする。 $b2$ は $v$ の子として新しい葉 $v_1, v_2$ を生成し,

$$\begin{aligned} f(t) &= v_1 && \dots \text{ 任意の } \tau \text{ の元 } t \text{ に対して} \\ f(t) &= v_2 && \dots \text{ 任意の } f^{-1}(v) - \tau \text{ の元 } t \text{ に対して} \end{aligned}$$

とする。 $b3$ は,  $p$ を親とし,  $c$ を子とするような新しい頂点 $v$ を生成する。 $b4$ は,  $p$ と $c$ に関して $CH$ のグラフの性質 $G2$ が満足されていないなら,  $p$ と $c$ を直接結ぶ辺を削除する。 $b5$ は,  $v$ の子が1つだけ存在し $G4$ に反する場合に,  $v$ とその子をまとめて1つの頂点とする。

## 5.2 アルゴリズム

新しい入力( $p \ z \ t$ )が与えられたときの,  $CMS$ による $CH$ の更新アルゴリズムについて述べる。 $CMS$ がある時点までに受け取る止の情報の集合を $P$ , 負の情報の集合を $N$ とする。また, その時点において $CH$ の表現する $Z \times T$ の部分集合を $S$ とする。 $CMS$ は,

$$P \subset S \subset N \quad \star$$

を満足するように $CH$ を更新する必要がある。(  $p \ z \ t$  )の入力直前には $\star$ が満たされていたと仮定する。さて,  $\star$ は,  $V$ の任意の頂点 $v$ に対して,

$$\Sigma P(g^{-1}(v)) \subset F(v) \subset (\Sigma N(g^{-1}(v))) \quad \star$$

が成り立つことと等価であることが,  $F(v)$ の単調性を用いて得られる。ただし,

$$\Sigma P(z) = \{t \mid (z', t) \in P \ \& \ z' \text{ は } z \text{ の子孫}\}$$

$$\Sigma N(z) = \{t \mid (z', t) \in P \ \& \ z' \text{ は } z \text{ の祖先}\}$$

である。

概念階層 $CH = \langle G, Z, T, f, g \rangle$ の特徴の1つは, 写像 $g$ だけを変化させることによって, 表現する集合 $S$ をある程度うまく変更できることである。 $g(z)$ を祖先の方向に移動すれば, 属性 $z$ を持つ対象の集合だけを大きくでき, 子孫の方向に移動すれば, 属性 $z$ を持つ対象の集合だけを小さくできる。この性質を有効に利用して,  $CH$ 更新において, 入力( $p \ z \ t$ )に対する $CH$ の更新が $g(z)$ の手直しだけで済む場合はなるべくその他のものを変化させないことにする。

新しい入力( $p \ z \ t$ )が与えられたとき, それによって,  $\Sigma P(g^{-1}(v))$ には,  $v$ が $g(z)$ の祖先ならば元( $z, t$ )が加わる。それ以外の場合は変化しない。また,  $\star$ の第2, 3項はまったく変化しない。従って新たに満足すべき条件は,  $g(z)$ の任意の祖先 $v$ に対して,

$$(z, t) \in F(v) \subset (\Sigma N(g^{-1}(v))) \quad \star$$

が成り立つことである。これは3つにわけられる。

$$(z, t) \in \Sigma N(g^{-1}(v)) \quad (5-1)$$

$$F(v) \cap \Sigma N(g^{-1}(v)) = \phi \quad (5-2)$$

$$(z, t) \in F(v) \quad (5-3)$$

このうち(5-2)は仮定によりすでに満足されている。

- (1) (5-1)が満足されていない時。(注意：ここでは(5-2)は変化しない)  
 $g(z)$ を祖先の方向に探索して(5-1)を満たす頂点を探索する。  
 見つかったら、その頂点を $v$ とする。  
 無い時には祖先の頂点の下に頂点 $v$ を生成する。  
 $g(z) = v$ とし、(2)へ。
- (2) (5-1)と(5-2)が満足されている時。  
 基本方針：(5-1)と(5-2)を保存したまま(5-3)を満たすように $G$ に変更を加える。  
 \* (5-3)を満足していればなにもしない。  
 \*  $g(z)$ を祖先方向に再び移動させて、(5-3)を満足させることを試みる。  
 成功すればend, 失敗すれば次へ行く。  
 \* 対象 $t$ だけを含む葉を生成し( $b1$ または $b2$ )、その葉を $g(z)$ の子に追加する。  
 これらの実行過程で、2つ以上の辺で到達可能な頂点を直接結ぶ辺(direct arc)が生じたり、1つの子しか持たない頂点を生み出す場合には、 $b4, b5$ を適用する。  
 入力 $t$ が負の例の場合、 $g(z)$ の探索は子孫の方向にし、類似の手続きによって更新がなされる。詳細は省略する。

## 6. 帰納的学習システムLS/1からの入力による学習

CMSは、帰納的学習システムLS/1の「知的サブシステム」として用いることができる<sup>2)3)4)</sup>。両者は、共に「情報」を受け取り、「知識」を作り出す。それは自分の仮説や推測を含んだものである。そして未知の「質問」を受け取った時「答え」を求めるのに用いられる。両者のちがいは、対象とする情報や知識の範囲と仮定の強さである。CMSは、1項述語というより狭い範囲を扱うが、階層性というより強い仮定を採用して構造化のスピードを上げる。CMSがLS/1の「知的サブシステム」として使われた場合、CMSはLS/1の情報処理を助ける「スペシャリスト」としての意味を持つ。帰納的学習システムLS/1において、

$Q = (\text{what is dog}), \quad A = (\text{mammal})$

$Q = (\text{what is dove}), \quad A = (\text{bird})$

という2つの問答がなされたとする。これから得られる関係情報は、「常識的」には、

(p mammal dog)

(p bird dove)

である。しかし、LS/1には初期知識として特定の言語知識や問題領域知識も与えていないので、上の問答から得られ、CMSの入力となるべき情報は図2のようなものである。ただし $\$$ と $\$$ は変数で、 $\$$ に対応する単語列が対象として取り出されている。

図2の入力による学習過程を説明する。CHの初期状態は、 $Z = T = \phi$ を満たせば、 $G$ は任意である。ここでは $G$ は存在しないとしておく。入力1においては $Z = T = \phi$ なので、対象(what)、属性( $(\$ \text{dog}) (\text{mammal})$ )は、それぞれ、 $Z, T$ に初めて出現することになる。このような場合、CMSは根の直下に頂点を生成して、そこにその属性、対象を対応させる。そして $Z, T$ にその属性、対象が追加される。これは入力2、3の場合も同様である。入力4の属性は初出であるが、対象は入力1で既に出現している。CMSはこの場合負の例がまだないので属性( $(\$ \text{dove}) (\text{bird})$ )を対象(what)が対応している葉 $n1$ に対応させる。入力5、6の属性、対象は初出だから根の直下

に頂点を生成してつける。入力7, 8では属性だけが初出だからそれぞれの属性を対象のある葉に対応させる。入力9では属性, 対象共に既出である。この時, CMSは同じ属性を持つ2つの対象をまとめるような頂点を生成し, そこにその共通する属性を対応させる。入力10でも同様である。最終的に生成されたCHは図3のグラフを持つ。

1	( p ( (\$ dog) (mammal)) (what) )	n0-*-n7-*-n3 [ (mammal) ]
2	( p ( (what \$) (mammal)) (dog) )	
3	( p ( (what dog) (\$) (mammal))	--n5 [ (bird) ]
4	( p ( (\$ dove) (bird)) (what) )	
5	( p ( (what \$) (bird)) (dove) )	*-n6-*-n2 [ (dog) ]
6	( p ( (what dove) (\$) (bird) )	
7	( p ( (what \$) (\$\$) (dove) )	--n4 [ (dove) ]
8	( p ( (what \$\$) (\$) (bird) )	
9	( p ( (what \$) (\$\$) (dog) )	--n1 [ (what) ]
10	( p ( (what \$\$) (\$) (mammal))	

図2 CMSへの入力情報

図3 作成されたCHのグラフ

## 7. むすび

概念管理システムCMSの概要について述べた。CMSのための知識構造として、概念階層(CH)と呼ばれる階層構造を準備し、その性質を調べ、さらにCMSによるCHの更新アルゴリズムを検討した。CMSが、帰納的学習システムLS/1の知的サブシステムとして動作し得ることを確認した。

今後の最大の課題は、入力が増大した時、どのようにして安定な階層構造をうまく抜き出していくかという問題を解決する枠組みを構築することである。また、帰納的学習システムLS/1からの要請としては、包含関係以外の手がかりを用いて階層構造の発見する方法や、n項関係の知識など他の知識構造を扱う知的サブシステムとの相互作用についても検討する必要がある。

## 文 献

- [1] 赤間 清, 小笠原 秀美: 帰納的学習システムの側面から見た概念の階層構造, 日本認知科学会, 第2回大会発表論文集 B-2 P23-24 (1985)
- [2] 赤間 清, 小笠原 秀美, 滝川雅己: 簡単な意味世界を帰納的に構成する試み, 昭和60年電気関係学会北海道支部連合大会講演論文集 p329-330 (1985)
- [3] 赤間 清: 知識の構造化を重視した学習のモデル, 情報処理学会, 知識工学と人工知能研究会資料, 45-5, P33-40 (1986)
- [4] 赤間 清, 滝川 雅己: 関係管理システム, 情報処理学会, 知識工学と人工知能研究会資料, 46-2, P9-16 (1986)