

TM-0170

仮説選定機構の一実現法

鶴 巻 宏 治 (NTT)

國 藤 進、古 川 康 一 (ICOT)

May, 1986

©1986, ICOT

ICOT

Mita Kokusai Bldg. 21F
4-28 Mita 1-Chome
Minato-ku Tokyo 108 Japan

(03) 456-3191~5
Telex ICOT J32964

Institute for New Generation Computer Technology

仮説選定機構の一実現法

國藤 進 (ICOT),

鶴巻宏治 (NTT複合通信研究所),

古川康一 (ICOT)

1. はじめに

人間の問題解決・推論過程の記号論的本質は、演繹 (deduction)・帰納 (induction)・発想 (abduction) [Kunifuji 86, Arikawa 86]にあるといわれている。従って、知識情報処理システムのプロトタイプ構築にあたって、演繹・帰納のみならず発想の過程を支援する機能を試作してみることは、極めて重要な研究課題である。しかしながら、従来、知識情報処理システムあるいは人工知能の研究で取り上げられていた研究課題は、主として演繹あるいは帰納による推論であり、発想に連なる推論機構研究はほとんど行なわれていない。そこで本論文では、発想推論 (abductive inference) に関する基礎研究として、特に仮説生成・検定という人間の最も良く使用する推論形式を取り上げ、その論理プログラミングによる試

作研究を通じて、発想推論の枠組を明らかにすることを主題に、簡単な実験研究を行なったので報告する。

さて人間の問題解決・推論過程の本質をバース哲学 [Yonemori 81] 風にみると、未知の観測事実を説明する仮説を生成する過程、得られた仮説から論理的に導出される帰結を得る過程、およびそのようにして得られた帰結を何らかの実験によって検証し、最初に生成した仮説を検定する過程からなる [Kunifuji 86]。このような観点からすると、発想とはある驚くべき事実（困難な未知の問題）に遭遇した人間が、その事実を説明する（その問題を解決する）仮説を直観的に得る人間の思考過程である。周知のように計算機科学での帰納推論は上述の仮説生成・検定過程からなるが、そこではある閉じた知識体系の中で極限としての同定 [Arikawa 86] をめざした仮説の生成を行なっている。しかしながら、本当に発想推論と呼ばれうるものでは、その知識体系に関する知識が不十分なので、人間は種々の知的技法を用いて、何らかの意味のある仮説を生成したり、合理的な仮説検定を行なっている。仮説生成・検定の仕方により、各種の発想推論図式が分類される。ここでは、そのうち仮説選定に

関する考察結果を述べる。

2. 発想と仮説生成・検定

パース [Yonemori 81, Ohsuga 86] によって指摘されているように、一般に仮説生成 (hypothesis generation) とは、経験的知識である観測事実 “O” と一般的知識である既知の公理 “H ト O” (ただし、“ト” は証明可能を意味する) とから、未知の知識である仮説 “H” を推測する記号処理過程である。すなわち、仮説生成を一階述語論理の用語法を用いて直観的に説明すると、次のような推論図式となる。

$$O \quad (1)$$

$$\frac{H \text{ ト } O}{\quad} \quad (2)$$

$$H \quad (3)$$

この推論図式を科学的理論の理論形成システムへ展開したのが、Theorist [Poole 85] である。Theoristでは図1に見られるように、与えられた観測事実の集合 “O” を説明する適切な仮説 (理論) “h” を、作り付けの可能な仮説 (理論) 集合 “H” の中から選択していく。すなわち Theoristでは、次のような推論図式を採用してい

る。

O 、 H 、 F を一階述語論理の節で表現された互いに独立な観測事実集合、前もって与えられた可能な仮説集合、およびその問題を解くのに利用可能な既知の知識集合とする。このとき、「仮説 h が観測事実集合 O を説明可能である」とは、次の場合、しかもその場合に限る。

$$\textcircled{1} \quad F \not\models O \text{ (ただし } F \not\models \square \text{)}、\text{かつ} \quad (4)$$

$$\textcircled{2} \quad h \subseteq H、F + h \vdash O \text{ (ただし } F + h \not\models \square \text{)}。 \quad (5)$$

ここに“ $\not\models$ ”は証明不可能を、“ $\square$ ”は矛盾を、“ $\subseteq$ ”は部分集合を、“ $+$ ”は和集合を意味する。式(4)は(無矛盾な)既知の知識集合 F のみからは観測集合 O が証明できないことを意味する。また式(5)は理論 h を仮説集合 H のべき集合 2^H から適切に選べば、(無矛盾な) F と h との和集合から O が証明できることを意味する。

上述の Theorist の枠組に対して、次のような疑問が著者らによって提案された。

(a) 知識ベース KB ($\equiv F + h$) の無矛盾性をどのようにして保持していくか。

(b) 可能な仮説集合 H から適切な仮説 h を選択する仮説選択の評価基準は何なのか。

- (c) 複数の仮説が選択されるとき、新たな観測事実の集合 O' を用いて、単一の仮説へ絞り込むための評価基準は何なのか。
- (d) (b), (c) の評価基準にもかかわらず複数の競合する仮説が生成されるとき、そのどちらを採用するかを決定するための評価基準は何なのか。
- (e) そもそも観測事実集合 O から作り付けでない仮説集合 H をどのようにして生成していくか。
- (f) 観測事実集合 O と既知の知識集合 F との対応付けをどのようにして発見していくか。

ここでは、(a) は無矛盾性管理問題、(b) は仮説選択問題、(c) は仮説同定問題、(d) は競合仮説解消問題、(e) は仮説生成問題、(f) を知識ベース選択問題として位置付ける。本論文では、概数の都合上、特に (b) と (c) に関する著者らの見解を述べる。著者らは既に、(a), (d) ~ (f) に関する研究を部分的に含む知識ベース管理の基礎技術として、知識同化・調節機構 [Kunifuji 85] やトランザクション管理機構 [Kitakami 85] をインプリメントしてきた。本論文では、その成果を活用した人間の問題解決過程支援のための仮説選定機構の一実現法

を述べる。そのような機構の設計思想と、その設計思想に基づく仮説選定機構の実現例を紹介する。

著者らの提案する問題解決支援システムのシステム構成は図2のようになっている。図2の各機能は、上記問題の解決を支援するために導入された。すなわち、問題(a)を解決支援するためには無矛盾性管理機構が必要であり、また問題(b)～(f)を解決支援するためには、与えられた問題向けの各種の知識調節機構の実現が必要である。ここに従来行なっていた知識同化機構[Kunifuji 85]は、現実世界からの知識(規則や事実)を人間というフィルタを経由して取り出し、それが正しいと仮定し、既知の知識ベースFに無矛盾かつ系統的に挿入する過程の管理機構として実現される。当然のことながら知識同化機構はその一部として無矛盾性管理機構を包含し、後者はしばしば矛盾型メタ知識と呼ばれるメタ知識ベースを用いて管理されている。また知識調節機構[Kunifuji 85, Kitakami 85]は現実世界から与えられる観測された事実が正しいと仮定し、それを説明する仮説を帰納推論アルゴリズムによって自動構築していく過程の管理機構として実現されてきた。さらにトランザクション管理機

構 [Kitakami 85] は人間というフィルタ経由の外から与えられた知識（規則や事実）が正しいと仮定し、それを知識同化機構を通して F に挿入するとき、その知識に関連するトリガ型メタ知識と呼ばれるデーモンが起動され、それに伴い知識調節機構が稼働し F が別の F' に更新される過程の管理機構として実現される。そこにおいては、あるトランザクションという単位での同化と調節の機構が同時に動くことが要請される。

周知のように、一階述語論理の節表現という平坦な知識表現と一階述語論理の定理証明機を前提とするとき、優れた性能は望めない。そこで一階述語論理の部分集合である適切な知識表現とそれ向けの推論エンジンの導入が期待される。たとえばフレーム構造型知識表現とそれ向けの推論エンジンを導入すれば、それなりの性能は達成できる。本論文ではこの問題の解決法について取り上げる。

3. 知識表現

3.1 特徴

本章では問題 (b) と (c) を解決するための仮説選定問

題向きの知識調節機構実現のための枠組を明らかにする。
Theorist研究に刺激を受けて、著者らは図3に示される
ような仮説の選定機構を実現した。本機構の特徴は、次
の3点にある。

- (1) 既知の知識集合Fや可能な仮説集合Hにis_a関係
に基づくフレーム構造型知識表現を導入し、平坦な知
識表現に基づく場合の性能の悪さを改良した。
- (2) フレーム構造型知識表現にとって自然な評価基準を
用いて、仮説を順次選択する機構を導入した。
- (3) 現実世界で新たに追加観測すべき事実を次々に提示
し、ユーザからのはい／いいえの答えをもとに、考え
られる複数の仮説を単一のものに合理的に絞り込むた
めの評価基準を明らかにした。

ここに(2),(3)は仮説選定機構の一部として実現され
ている。また後述するように、(3)はフレーム識別実験
とフレーム確認実験の二種類に分類される。

3.2 フレーム型知識表現

与えられた問題を一階述語論理の枠組で明確に取り扱
うために、次のようなスロット対(三 “<SLOT-NAME>: s

lot-value ”) からなるフレーム型知識表現 [Poole 86-2] を採用する。ただし問題を単純化するために、ループを持たない階層的フレーム構造を考える。

```
[ FRAME <frame-name>
  hypothesis_of: <hypothesis-name>;
  <SLOT-NAME>1 : slot-value1 ;
  :
  <SLOT-NAME>n : slot-valuen ]
```

(6)

ここにスロット名の区別により、スロットの型は hypothesis-of 関係、is-a 関係と、それ以外の特性を表わす property-of 関係の三種に分類される。このフレーム型知識表現はスロット型に応じて、それぞれ次のような一階述語論理の形に変換・解釈されるものとする。一般に hypothesis-of 関係とそれ以外の関係 [Hayes 79, Poole 86-2] は、それぞれ次のように変換される。

$$\forall X(\text{hypothesis_of}(X, \text{<hypothesis-name>}) \Rightarrow \text{is_a}(X, \text{<frame-name>}))$$

(7)

$$\forall X(\text{is_a}(X, \text{<frame-name>}) \supset \langle \text{SLOT-NAME} \rangle_i (X, \text{slot-value}_i))$$

(8)

ここに“ $\Rightarrow$ ”は“ $\supset$ かつ $\subset$ ”の意味である。

[例題 1] 次のようなフレーム型知識表現を考える。

```
[ FRAME alkaline-earths
  hypothesis-of: アルカリ土類金属;
  is-a: metal;
  precipitate: 'SO4 -ion']
```

(9)

上述のフレームは、次のような一階述語論理の節表現に変換される。

$$\forall X(\text{hypothesis_of}(X, \text{アルカリ土類金属}) \\ \exists \text{is_a}(X, \text{alkaline_earths})) \quad (10)$$

$$\forall X(\text{is_a}(X, \text{alkaline_earths}) \supset \\ \text{is_a}(X, \text{metal})) \quad (11)$$

$$\forall X(\text{is_a}(X, \text{alkaline_earths}) \supset \\ \text{precipitate}(X, \text{'SO}_4 \text{ -ion'})) \quad (12)$$

4. 仮説選定問題

4.1 仮説選択問題

与えられた観測事実 O に対して利用可能な既知の知識集合 F が固定され、そのフレーム名が $frame$ として与えられているものと仮定する。 $frame$ に出現する全仮説名の集合を H_N とするとき、一般に仮説集合 H_X は次のよ

うにして与えられる。

$$H_X = \{ \text{hypothesis_of}(X, N) \mid N \in H_N \} \quad (13)$$

与えられた観測 O に対しては、 X は特定の x に例示されていると考えられるので、

$$H_x = \{ \text{hypothesis_of}(x, N) \mid N \in H_N \} \quad (14)$$

を考えれば十分である。簡単のため H_x を H と略記すると、与えられた frame 内で選択可能な仮説 h とは H のべき集合の元として定義される。

$$h \in 2^H \quad (15)$$

$|h| = 1$ のものを単純仮説、 $|h| > 1$ のものを複合仮説と呼ぶ。

さて F のうち hypothesis_of 関係の部分、 is_a 関係の部分、 property_of 関係の部分を一階論理の節表現したものを、それぞれ F_h 、 F_i 、 F_p とする。すると、次式が成立する。

$$F \equiv F_h + F_i + F_p \equiv F_h + F_{ip} \equiv F_{hi} + F_p \quad (16)$$

著者らの目標は仮説選択や仮説同定に関する合理的な評価基準を明らかにすることである。そのために、まず可能な仮説集合 2^H に hypothesis_of 関係と is_a 関係に基づく順序関係 “ $>$ ” を導入する。

$$h_1 > h_2 \equiv "h_1 \in 2^H, h_2 \in 2^H, \\ F_i + h_2 \vdash h_1" \quad (17)$$

この順序関係“>”によって、仮説空間 2^H は半順序束をなす。この関係“>”の意味で順序の大きい方を上位の仮説と呼ぶ。著者らの方針は、この順序関係の大きいものから順次仮説を選んでいこうというものである。一般に仮説の間の順序関係を定義(17)に従って調べることは無駄が多い。仮説空間に階層構造を導入することによって、その点が極めて単純化される。

さて一般に、観測 O は property-of 関係の事実として与えられる。従って、複数の仮説 h_1 と h_2 が観測事実 O を説明し、

$$h_1 \in 2^H, F_{ip} + h_1 \vdash O \quad (18)$$

$$h_2 \in 2^H, F_{ip} + h_2 \vdash O \quad (19)$$

しかも

$$h_1 > h_2 \quad (20)$$

とするとき、 h_2 を h_1 より冗長な仮説と呼ぶ。すると、冗長でない仮説を求めることが問題となる。幸いなことに、この問題は冗長でない根源仮説（始祖頂点）の集合を求める問題に帰着される。従って、著者らによって開

発された複合頂点間の到達可能性解析 [Kunifuji 79] によって解決可能である。

[例題 2] 図 4 のようなフレーム型知識表現が与えられているとする。このとき、“mortality(socrates, mortal)” という観測事実が与えられたとする。この観測事実を説明する可能な仮説として、このフレームでは “ $\forall X(is_a(X, living_thing) \supset mortality(X, mortal))$ ” という特性の継承がなされるので、“hypothesis_of(socrates, 男)” から “hypothesis_of(socrates, 生物)” にいたる 9 種類の可能な仮説が生成される。本例題では、最も自然な仮説として最上位のフレーム名が選択される。

4.2 仮説同定問題

次に仮説同定に関する評価基準を明示しよう [Poole 86-1, Tsurumaki 85, Seki 85]。複数の仮説 h_1 と h_2 が観測事実 O を説明したとする。

$$F_{ip} + h_1 \vdash O \quad (21)$$

$$F_{ip} + h_2 \vdash O \quad (22)$$

新たな観測事実 O' に関する実験を行ない、

$$F_{ip} + h_1 \vdash O' \quad (23)$$

$$F_{ip} + h_2 \neq 0 \quad (24)$$

となるならば、当然のことであるが仮説 h_1 を採用することになる。このことを若干形式的にいうと、次のようになる。ある仮説 h を仮定するということは、そのフレームに関する property-of 関係に関する推論規則 (8) が使えることになる。この property-of 関係には、仮説同定という観点からすると二種類のものがある。最初に同定すべき property-of 関係は (23), (24) の関係をみたす特性のユーザへの問い合わせ実験である。これにより、どちらかの仮説をふるいおとすことができる。残りの property-of 関係に関するユーザへの問い合わせ実験は、与えられた仮説が正しいことを確かめるための実験である。以下で紹介する試作中の実験システムは前者のみ実働化している。

4.3 仮説選定の手続き

前節の問題提起に従って設計した仮説選定手続きを組み込んだ実験システムを試作した。その処理フローの概略を図5に示す。図5に示されているように、式 (18) ~ (20) の条件を満足するように仮説を生成している。

システムは、任意の二つの仮説 h_1 、 h_2 について、“ $h_1 > h_2$ ” なる順序が付けられているものについては必ず h_1 から先に後述の explain 述語による証明可能性の評価を行なう。同時に選択した仮説が既に証明に成功した仮説に対して、冗長かどうかの判断を行なっている。すなわち関係 “ $>$ ” に従って生成される仮説は、観測事実を説明し得る範囲でより上位のものから生成される。

同定プロセスでは、選択した仮説を識別するための実験が行なわれる。仮説が正しいと仮定したとき新たに観測されるはずの特性について、システムがユーザーに問い合わせる。前述のフレーム識別実験に相当する問い合わせ全てについて、ユーザーの確認が得られたとき、その仮説が正しいとする。

5. 実現法

5.1 Prolog 向き知識表現

システムの実働化の現状について述べる。Prolog で簡潔に実現する便宜上、次のような知識表現を採用した。後述する実行例で用いた知識を例にとって説明する。

システムの持つ知識は、フレーム構造型知識表現 F に
 よって記述されている。hypothesis-of 関係、is-a関係、
 property-of 関係は、それぞれ次のように与えられる。

hypothesis_of(FrameName, Hypothesis). (25)

is_a(FrameName, SuperFrameName). (26)

property_of(FrameName, HornClause). (27)

ここに、FrameName はフレーム構造 F における位置付
 けを示すためのフレーム名である。Hypothesisには仮説
 名に対応する素命題が代入される。

〔例題 3〕 未知のイオン同定問題という後述する実行
 例を用いて説明する。例えば、

hypothesis_of('Zn', zn(X)). (28)

is_a('Zn', transition_element). (29)

property_of('Zn', (precipitate_s_ion(X):-
 zn(X), acidity(X))). (30)

のように記述する。(30)式は、『任意の(水溶液) X 中
 に亜鉛イオンがあり、かつ、X が酸性であれば、水溶液
 X は硫黄イオンと反応して沈澱を生ずる』という知識を
 表している。

5.2 システムの処理

システムは、説明すべき観測事実が与えられると、仮説選定の評価基準を満足する仮説を次々に選択していく。そして、知識集合 F の知識と、いま選択した仮説とから与えられた観測事実が証明可能かどうか調べる。生成される仮説は、仮説集合 H の要素の連言に制限されているので、 2^H の元で与えられる。しかし、仮説集合も階層構造化されていることと、冗長な仮説は認めないという制限から、実際は選択される可能性のある仮説の数はもっと少ない。何故なら、 h_1 で説明できることが分かったら " $h_1 > h_2$ " なる h_2 の生成は有り得ないからである。

選択過程で複数の候補が得られた場合は、同定過程でユーザと対話的な選択／確認を行なう。その際、システムはうまく識別できるための質問を見い出さなければならない。 F の規則のヘッド部が、質問となる素命題となり得るわけだが、現在は F をトップダウン、横型優先探索を用いて、この質問を見付け出している。ここにシステムは (23), (24) 式の O^+ を見付けている。

図 6 で示したのは、観測された（互いに独立な）事実

集合が、選択した仮説と知識集合 F_{ip} とから証明可能かどうかを調べるプログラム `explain` 述語の概略である。この述語は、メタプログラミング技法 [Kunifuji 85-1] によって実現している。ここでは、対象知識として `Prolog` の内部データベース自身を用いているために、引数は次の3つである。第1引数は、証明されるべきゴールとなる観測事実を与える。第2引数は、選択した仮説のフレーム名を、複合仮説の場合に対応するためリストとして与える。第3引数には、フレーム名に対応する仮説名をリストで与える。第2,3引数に与えた仮説で、第1引数の観測事実が証明可能であるとき、`explain` 述語は成功する。

`explain` 述語の第1節は、`true` が常に証明可能であることを示す。第2節は、`ObsH` と `ObsI` の連言として与えられた観測事実が仮説 `Hypothesis` で証明可能であるためには、`ObsH, ObsI` それぞれが同じ仮説 `Hypothesis` で証明可能でなければならないことを示す。第3節は、証明しようとするゴールである観測事実 `Obs` が、選択した仮説のみから証明できる場合を示す。サブゴールとなっている述語 `explain _ hyp` は、`Obs` が `Hypothesis` に含まれるか、

これより上位のフレームの仮説名であるとき成功する。
第4節は、上の3つの節が成立せず、選択したフレームに属するかあるいは継承される特性を表わすホーン節のヘッドとゴールObsが単一化可能である限りにおいて、そのホーン節のボディを新たなゴールとして証明可能であることを調べればよいことを示している。

6. 実行例

本システムを水溶液中の未知のイオンの診断過程に応用した例を示す。知識集合は図7に示した例を用いる。

図8は、観測事象として、

1-1) 水溶液Xに炭酸イオンを加えると沈澱を生じた；

1-2) 水溶液Xに硫酸イオンを加えると沈澱を生じた、

という二つの観測事象から水溶液X中に含まれるイオンを推測する例である。

図中*で示されているのが、1-1), 1-2)を説明する全ての仮説である。ここでは、“「水溶液Xにはアルカリ土類金属イオンが存在する」；「水溶液Xには鉛イオンが存在する」”という2種類の仮説を生成している。

システムは、これらの候補を絞り込むために質問を

してくる。これに対し、ユーザはyes かnoで答える。このマンマシン・インタラクションによって得られた情報により、システムは仮説を選択し、確認していく。この例の後半では、「水溶液Xは水酸基イオンで沈澱を生じるか」という質問に対してはnoと答えている。システムは、この情報によって、「Xに存在するイオンはアルカリ土類イオンである」という仮説を提示している。

図 9の例では、以下の2つが観測事象となっている。

- 2-1) 水溶液Xは硫酸イオンで沈澱を作った；
- 2-2) 水溶液Xは水酸基イオンで沈澱を作った。

図中*で示されているのが、2-1), 2-2)を説明する全ての仮説である。この場合は、“「水溶液Xにはカルシウムイオンが存在する」；「水溶液Xには鉛イオンが存在する」；「水溶液Xにはアルカリ土類金属イオンと遷移元素イオンが存在する」”の3種類の仮説が選択されている。

次にシステムは「水溶液Xは硫酸イオンで沈澱を生ずるか」という質問をしてきている。ユーザはこれに対してyes と答え、その結果、第1の仮説は棄却されている。さらにシステムは「水溶液Xはハロゲンイオンで沈澱を

生ずるか」という質問をしてきている。これに対してユーザがnoと答えることによって、システムは第3の仮説「水溶液Xにはアルカリ金属イオンと遷移元素イオンが存在する」を提示している。

7. 暗黙推論との関係

知識集合Fに構造化知識を持つ以上、当然、例外知識も記述したくなる。これに対して、次のような方法が考えられる。

一般にproperty-of関係は、(*で表記される)暗黙値をもつものともたないもので変換形式が異なり、前述のように後者は形式(8)で変換される。前者はメタオペレータMを用いて、形式(31)のように変換される。このメタオペレータMは(32)に示されているように、Reiterの正規暗黙値オペレータM[Reiter 80, Poole 86-1]]と密接な関係がある。

$$\text{is_a}(X, \langle \text{frame-name} \rangle)$$

$$:M(\langle \text{SLOT-NAME} \rangle_i (X, \text{slot-value}_i))$$

$$\therefore \langle \text{SLOT-NAME} \rangle_i (X, \text{slot-value}_i) \quad (31)$$

$$\begin{array}{l}
\text{is_a}(X, \langle \text{frame-name} \rangle) \neq \\
\text{not}(\langle \text{SLOT-NAME} \rangle_i (X, \text{slot-value}_i)) \\
\hline
\therefore \langle \text{SLOT-NAME} \rangle_i (X, \text{slot-value}_i) \quad (32)
\end{array}$$

[例題4] 図4で例外知識を許すとき、次のような変換を与えればよい。

$$\begin{array}{l}
\forall X(\text{is_a}(X, \text{bird}) \wedge M(\text{can_fly}(X, \text{sky}))) \\
\qquad \qquad \qquad \supset \text{can_fly}(X, \text{sky}) \quad (33)
\end{array}$$

式(33)は、次のように解釈することもできる。

$$\begin{array}{l}
\text{is_a}(X, \text{bird}) \neq \text{not}(\text{can_fly}(X, \text{sky})) \\
\hline
\therefore \text{can_fly}(X, \text{sky}) \quad (34)
\end{array}$$

ここに一般にReiterのMオペレータの実現は非単調論理になるので、その取り扱いが難しい。著者らはMオペレータを、たとえば次のように制限することによって取り扱うことにする。

$$\begin{array}{l}
\text{can_fly}(X, \text{sky}) :- \text{is_a}(X, \text{bird}), \\
\qquad \qquad \qquad \backslash +(\text{cannot_fly}(X, \text{sky})). \quad (35)
\end{array}$$

$$\text{cannot_fly}(\text{ostrich}, \text{sky}). \quad (36)$$

このようにPrologの否定“\+”を積極的に利用し、例外値を明記するもののみ許容することによって、いわゆる暗黙推論をも利用可能になる。以上述べてきた考え方はProlog上での実現法を与えるが、これを厳密に議論

するには、Mオペレータを正当化する非単調論理の導入が必要となる。

8. おわりに

本稿では、Poole らのTheoristシステムの理論を参考にして、仮説選定を行なうシステムの構築について報告した。そこでは、可能な仮説の集合である仮説集合が階層的にフレーム構造化されていることが特徴である。具体的には複数の仮説の中からより適切な仮説を選定するための評価基準を明らかにするための実験ツールを作った。また、仮説の同定には、ユーザとシステムとの対話的な要素が不可欠であるとし、システムには対話的に判断するモジュールを導入した。本稿で取り上げた2つの評価基準の根拠となる概念は、フレーム構造にとって自然な上位の仮説という概念、およびそれから必然的に導かれる非冗長な仮説という概念である。また、仮説の同定には、ユーザとシステムとの対話的な要素が不可欠であるし、システムには対話的にユーザと対話するモジュールを導入した。

今後の課題として、次のようなものが挙げられる。

- 1) 仮説を選定していく過程上で、新たに得られた知識を知識ベースに無矛盾に獲得する機能を追加する。
- 2) 一般に仮説は選択・同定を経るとその確実さが増す。その結果、それは信念として一般的知識に組み込まれる。このような知識はいわゆる truth maintenaceの対象となる。

[謝辞] 本研究の機会を与えていただきましたICOT測一博研究所長およびNTT 複合通信研究所小森和昭室長および熱心に議論していただいたGoebel助教授 (Univ. of Waterloo)およびICOT第1研究室の世木博久、武脇敏晃両氏に感謝致します。

[参考文献]

[Arikawa 86] 有川節夫：帰納推論と類推－理論と応用－、測一博監修知識情報処理シリーズ第2巻“知識の学習メカニズム”、共立出版、1986.

[Goebel 86] Goebel, R., Furukawa, K. and Poole, D.: Using Definite Clauses and Integrity Constraints as the Basis for a Theory Formation Approach to Diagno

stic Reasoning, to appear in International Conference on Logic Programming, Imperial College, London, July 1986.

[Hayes 79] Hayes, P.: The Logic of Frames, in Reading in Artificial Intelligence (Webber, B. and Nilsson, N. eds.), Tioga, 1979.

[Kitakami 84] Kitakami, H., Kunifuji, S., Miyachi, T. and Furukawa, K.: A Methodology for Implementation of a Knowledge Acquisition System, Proc. of the 1984 International Symposium on Logic Programming, Atlantic City, U.S.A., Feb. 6-9, 1984.

[Kitakami 85] 北上 始, 國藤 進, 宮地泰造, 古川 康一: 論理プログラミング言語 Prolog による知識ベース管理システム, 情報処理, Vol. 26 No. 11, 1985.

[Kunifuji 79] 國藤 進, 竹島 卓: 複合頂点間の到達可能性解析, 電子通信学会 CAS79-110, 1979年10月.

[Kunifuji 85-1] 國藤 進, 北上 始, 宮地泰造, 古川 康一, 知識工学の基礎と応用 [第4回] - Prolog における知識ベースの管理 -, 計測と制御, Vol. 24 No. 6, 1985.

[Kunifuji 85-2] 國藤 進, 武脇敏晃, 世木博久, 竹内彰一, 大木 優, 古川康一, 鶴巻宏治: メタプログラミングによる論理プログラミングと知識情報処理の融合, 第28回自動制御連合講演会前刷“特別講演”, 国立教育会館, pp. Ⅲ-1~Ⅲ-10, 1985年11月7日.

[Kunifuji 86] 國藤 進: 演繹・帰納・発想の推論機構化をめざして、淵一博監修知識情報処理シリーズ第2巻“知識の学習メカニズム”, 共立出版, 1986.

[Miyachi 84-2] Miyachi, T., Kunifuji, S., Kitakami, H. and Furukawa, K.: A Knowledge Assimilation Method for Logic Databases, New Generation Computing, 2-4, 385/404, 1984.

[Ohsuga 86] 大須賀節雄: 知的情報処理の現状と課題、計測と制御、Vol.25, No.4, pp.299-307, 1986年4月.

[Reiter 80] Reiter, R.: A Logic for Default Reasoning, Artificial Intelligence, Vol.13, pp.81-132.

[Poole 85] Poole, D., Aleliunas, R. and Goebel, R.: Theorist: a logical reasoning system for defaults and diagnosis, submitted as a chapter in the volume Knowledge Representation, N.J.Cercone & G.McCalla

(eds.), IEEE Press, in preparation.

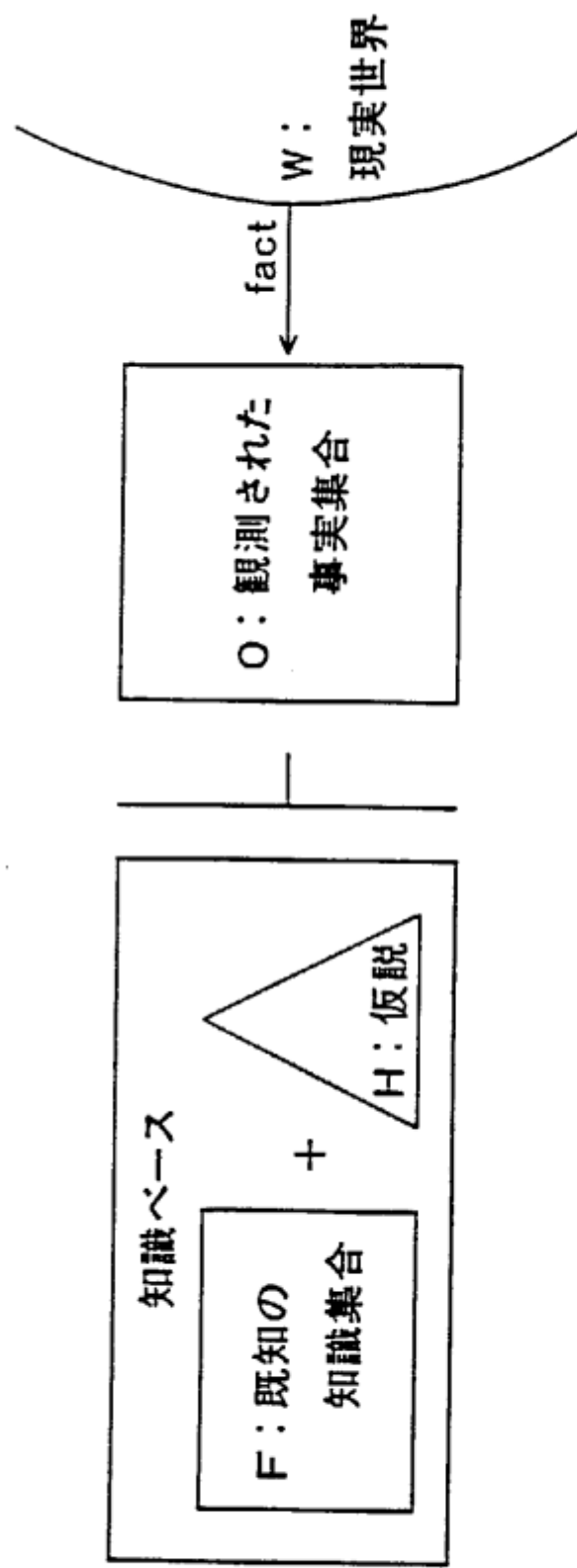
[Poole 86-1] Poole, D.L.: Default Reasoning and Diagnosis as Theory Formation, University of Waterloo, Technical Report CS-86-08, March 1986.

[Poole 86-2] Poole, D.L. and Goebel, R.: Frame Selection based on Defaults and Theory Preference, private communication, April 1, 1986.

[Seki 86] Seki, H. and Takeuchi, A: Integrity Constraints as Criteria for Theory Formation / Selection, to appear in ICOT TR, March 1986.

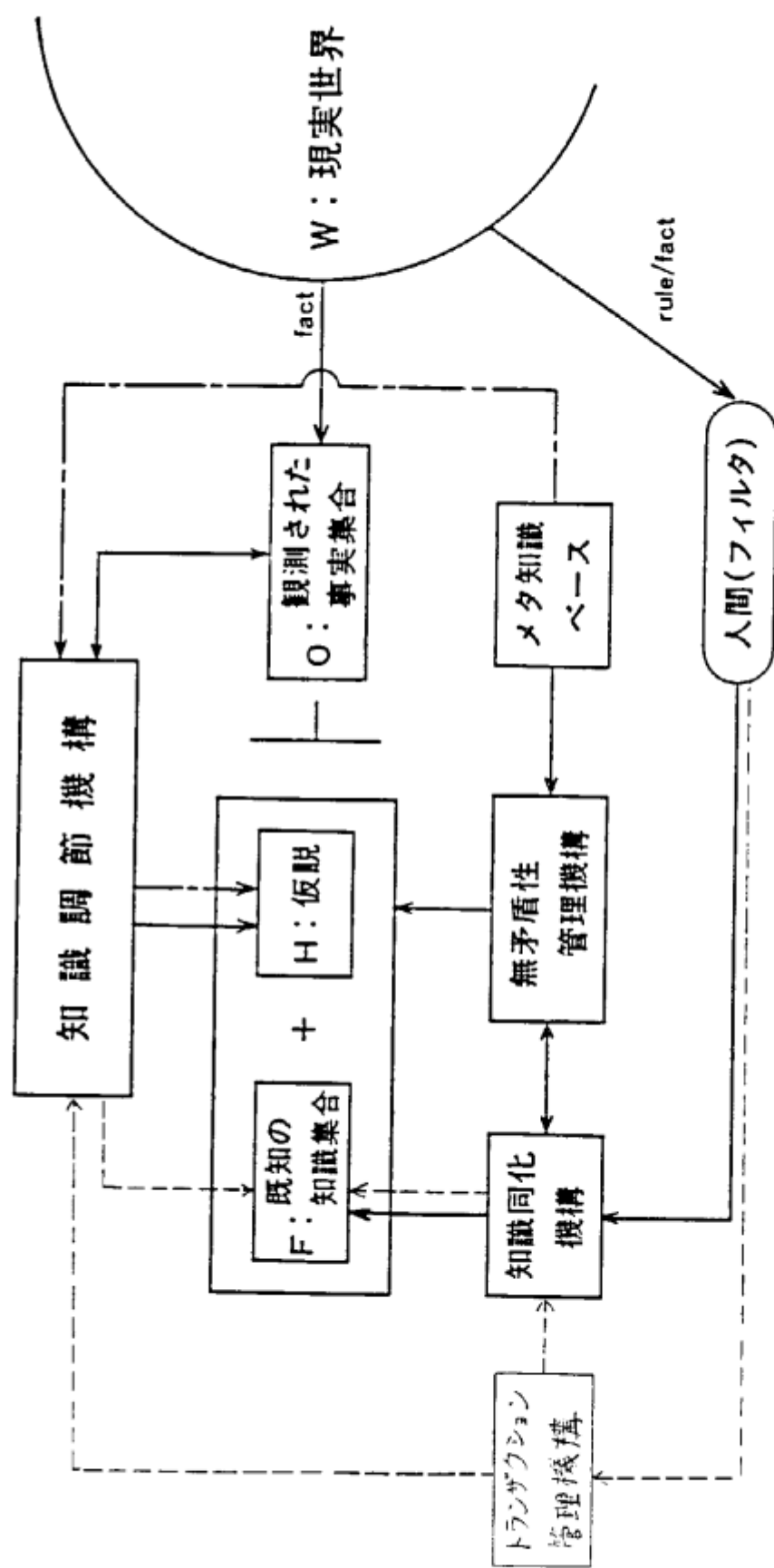
[Turumaki 85] 鶴巻宏治, 國藤 進, 古川 康一: メタプログラミングによる類推システムの試作について、日本ソフトウェア科学会第2回大会論文集, 1985年11月.

[Yonemori 81] 米盛裕二: パースの記号学、勁草書房, 1981.



1. Fは矛盾していない。
2. FだけからOを説明できない。
3. Fに矛盾しないある仮説Hを考えると,
FとHからOが説明できる。

図1 仮説推論の枠組



2. [x]

仮説推論と知識獲得

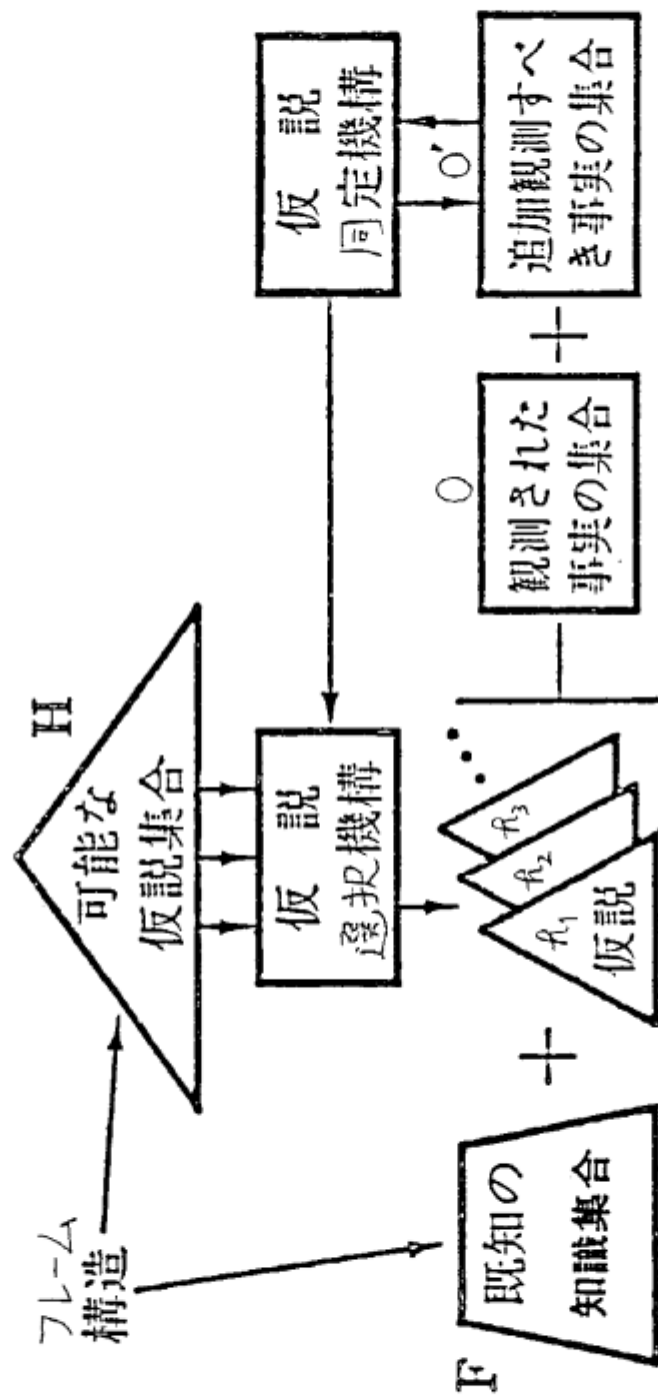


図 3 仮説選定機構の枠組

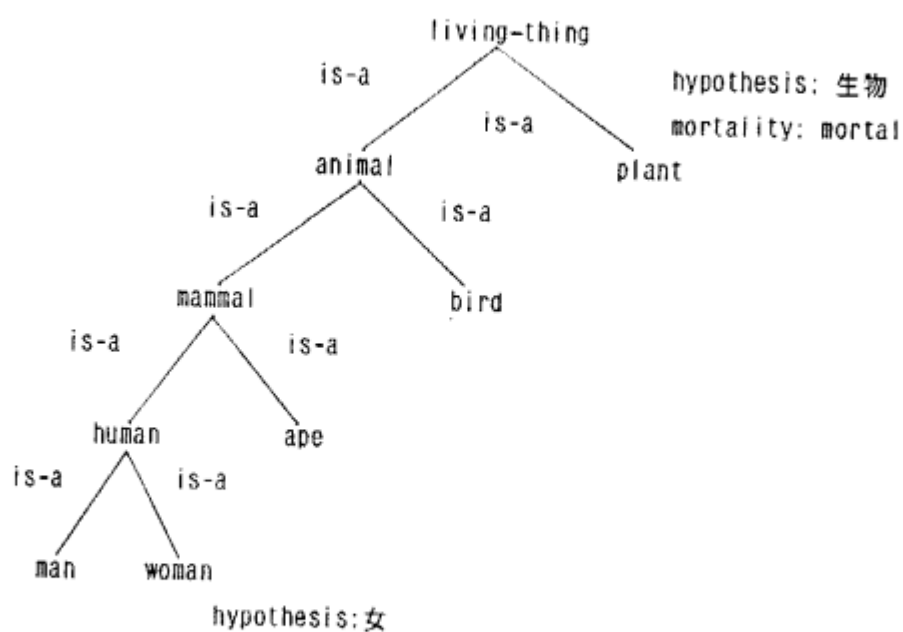


図4 生物フレームの知識表現

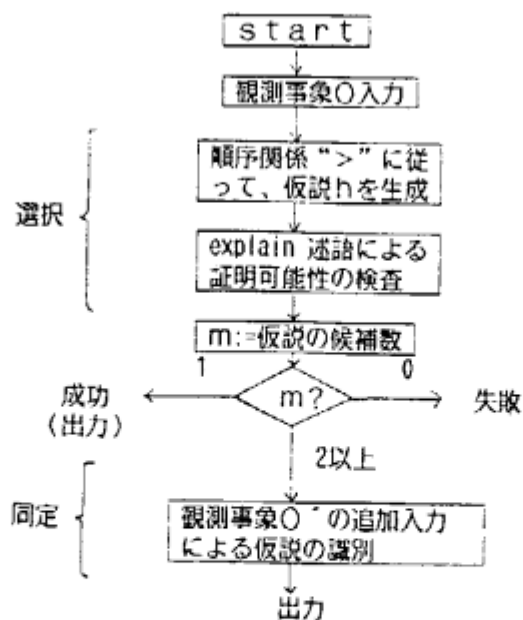


図5 仮説選定の処理フロー

```

explain(true,_,_) :- !,true.

explain((ObsH,ObsT),Frame,Hypothesis) :- !,
    explain(ObsH,Frame,Hypothesis),
    explain(ObsT,Frame,Hypothesis).

explain(Obs,Frame,Hypothesis) :-
    explain_hyp(Obs,Frame,Hypothesis),!.

explain(Obs,Frame,Hypothesis) :-
    explain_prop(Obs,Frame,Body),
    explain(Body,Frame,Hypothesis).

explain_prop(Obs,[FrameH|_],Body) :-
    ( property(FrameH,(Obs:-Body))
    ; inherit_property(Frame,(Obs:-Body)) ).

explain_prop(Obs,[_!FrameT],Body) :-
    explain(Obs,FrameT,Body).

inherit_property(Frame,Property) :-
    super_concept(Frame,SuperFrame),
    property(SuperFrame,Property).

```

図 6 explain プログラム


```

| ?- start.
observations : precip_CO3_ion(x),
observations : precip_SO4_ion(x).
.
.
.

Hypotheses for (precip_CO3_ion(x), precip_SO4_ion(x)) is
alkaline_earth exist_alkaline_earth(x) ; ----- *
Pb exist_Pb(x) ----- *
```

Hi, my sweet! Give an answer to me about the following question with 'y' or 'n'.
Is "precip_OH_ion(x)" right ?
>> n

```

Rejected Hypotheses...
    Pb exist_Pb(x)

Verified Hypotheses...
    alkaline_earth exist_alkaline_earth(x)
```

yes
| ?-

图 8 实行例 1

```

| ?- start.
observations : precip_SO4_ion(x),
observations : precip_OH_ion(x).
:
:
Hypotheses for (precip_SO4_ion(x), precip_OH_ion(x)) is
    Ca exist_Ca(x) ; ----- *
    Pb exist_Pb(x) ; ----- *
    alkaline_earths exist_alkaline_earths(x) & transition_element exist_transition_e
lement(x) ----- *

Hi, my sweet! Give an answer to me about the following question with 'y' or 'n'.
Is 'precip_S_ion(x)' right ?
>> y

Rejected Hypotheses...
    Ca exist_Ca(x)

Hi, my sweet! Give an answer to me about the following question with 'y' or 'n'.
Is 'precip_halogen_ion(x)' right ?
>> n

Rejected Hypotheses...
    Pb exist_Pb(x)

Verified Hypotheses...
    alkaline_earths exist_alkaline_earths(x) & transition_element exist_transition_elem
ent(x)

yes
| ?-

```

図 9 実行例 2