

TM-0487

仮説推論に対する期待とイメージ
—昭和62年度知識システムシエルワーキンググループ・
—仮説推論サブワーキンググループ報告書—

井上克巳, KSS-WG・HYR-SWG

March, 1988

©1988, 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

仮説推論に対する期待とイメージ

—昭和62年度知識システムシェルワーキンググループ・
仮説推論サブワーキンググループ報告書—

井上 克巳（編集）

KSSワーキンググループ・HYR-SWG
（石塚 満〔東京大学〕：主査）

I C O T

昭和63年 3月31日

序

昭和62年度のI C O T知識システムシェルワーキンググループ(K S S - W G)は次世代知識システム構築のための5つの要素技術(仮説推論、問題解決モデルと知識表現、分散協調問題解決、知識獲得支援、設計型エキスパートシステム)についてサブワーキンググループ(S W G)を開催する形で運営された。本報告書はこのうち仮説推論S W Gに関係する。

現在I C O Tにおいて次世代ツールの研究開発の一つのテーマとして仮説推論(hypo-thetical reasoning; assumptio-based reasoning)の研究を行っている。またこれに関連してI C O Tの再委託研究としても仮説推論を課題に挙げているメーカーが多い。本サブWGはこうした研究のニーズに対応するべく、推論機構の一提案を目的として設置された。仮説推論は従来の古典論理の枠組みだけでは扱えなかった不完全な知識を扱うための一方式であり、真偽が不明な知識を取り敢えず真と仮定して推論を進める。後に矛盾する状況が起こったときには基になる仮説を修正することから非単調推論を実現する必要がある。従来の仮説推論の研究はより基礎的な推論メカニズムの立場から行われてきており、実際の問題解決においてどのように使われるかという観点からの研究は手がつけられていなかった。このため、本S W Gでは特にこの仮説推論を利用して問題解決を行うための枠組みについての考察を重視した。特にI C O Tにおいて次世代知識システムのための基盤ソフトウェア・ツールとして検討を進めている仮説推論システム(仮称・A P R I C O T)の構成にS W Gでの検討結果を反映させたいと考えている。

仮説推論サブWG(H Y R - S W G)は、仮説推論に関連する研究を行っている大学等の外部有識者、仮説推論を取り入れたエキスパートシステムの研究開発に取り組んでいるメーカーの研究者、及びI C O Tの研究者をメンバーとして構成された。外部有識者の方々には主として基礎理論面から論理に基づいて考察することにより、推論機構の具体化のための検討をお願いした。また、メーカーの方々については手持ちの問題を通して仮説推論の必要性や具体的な適用方式についての討論を担当していただいた。

本報告書はH Y R - S W Gの活動報告書と、各委員・オブザーバに自由な形式で執筆してもらった「仮説推論に対する期待とイメージ」を一つの報告書としてまとめたものである。この「期待とイメージ」は本S W Gにおける検討結果が反映されたものとなっている。本報告書がA I研究に対して一つの提言を投げ掛けられれば幸いである。

また、本報告書の出版に当たり、本研究の機会を与えてくださったI C O T研究所の淵一博所長、原稿執筆にご協力をいただいた石塚満主査を始めとする委員・オブザーバの方々、ならびにS W Gの活動・運営に関してご協力および助言をいただいたI C O T第5研究室の藤井室長および研究員の方々に感謝する。

昭和63年3月31日

I C O T 井上 克巳

目次

1.	昭和62年度KSS-WG・仮説推論SWG 活動報告書	5
	石塚 満 主査 (東京大学 生産技術研究所)	
2.	次世代知識ベース・ソフトウェア・アーキテクチャへのアプローチ	13
	石塚 満 (東京大学 生産技術研究所)	
3.	仮説推論を利用する枠組み	23
	中川 裕志 (横浜国立大学 工学部電子情報工学科)	
4.	仮説推論システムのイメージ	25
	久野 禎子 (東京工業大学 理学部情報工学科)	
5.	仮説推論を用いた問題解決	29
	井上 克巳 (ICOT 第5研究室)	
6.	ATMSの設計型問題への適用	31
	飛鳥井 正道 (キャノン 情報システム研究所)	
7.	VLSI論理設計における仮説推論	35
	丸山 文宏 (富士通研究所 システム研究部)	
8.	機械設計における仮説推論	39
	今村 聡 (機械技術研究所 システム部)	
9.	工作機械の主軸設計への仮説推論の適用	43
	加藤 文之 (松下電器産業 東京研究所)	
10.	モデルに基づく診断と仮説推論の関係	47
	赤埴 淳一 (NTT 情報通信処理研究所)	
11.	故障診断への仮説推論の適用	49
	和田 慎一 (日本電気 C&Cシステム研究所)	
12.	仮説推論のツール・イメージ	53
	道藤 静一 (三菱電機 中央研究所)	

1. 昭和62年度KSS-WG・ 仮説推論SWG 活動報告書

石塚 満 主査
(東京大学 生産技術研究所)

1. はじめに

次世代知識システムのツールの機能と構成について検討するKSSワーキンググループの中で、特に仮説推論を中心課題にしたのが本仮説推論(HYR)SWGである。

仮説推論は問題解決の際に、不明であり不足しているデータ/知識をとりあえず真として(仮説をたて)推論を進める推論形式である。仮説は仮説の素が存在する場合と、無から仮説を生成する場合(Abductionと称する)とがある。仮説は一種の不完全な知識であり、使用に際しては制約条件による無矛盾性のチェック、また真なる事実が判明したときには信念の翻意(belief revision)、非単調推論(non-monotonic reasoning)などの制御を必要とする。仮説推論によるゴール指向の問題解決を通し、生成された無矛盾な仮説の集合を解として求めることにより、従来のヒューリスティックな知識を重視した問題解決とは異なる、故障の診断や計画問題、設計問題等への応用も拓けつつある。

本SWGは9名の委員(大学より3名、公立研究機関より1名、企業より5名)とICOT第5研究室の担当者、オブザーバ若干名(NTTからなど)により活動し、本年度7回のSWGを開催した。仮説推論の要素技術、実用化技術、今後の知識ベース・システムにおける役割や姿などについて、論文のサーベイや各委員の意見を基にして討論し、理解を深め、今後の方向を探索した。

2. 活動方針

仮説推論に関しては、ATMSのように実用化されつつ技術がある一方、次世代知識ベースの基盤技術としての論理的枠組み上での仮説推論の諸課題がある。更に実用システムにどのように適用していくのかも、実例に則して検討し、必要とされる機能を抽出していく必要がある。参加していただいた委員には、仮説推論についてそれぞれが関心のある課題を担当してもらい、関連研究のサーベイや、自らの研究開発の内容、意見を述べ合って討議を深め、今後の方針を定める形で進めた。

必ずしも当初から分類した訳ではないが、議論した内容はおおまかに次の4項目に分類できる。

- a) TMS, ATMSを中心とする実用技術としての仮説推論とその高速化
- b) 論理の枠組み上での仮説推論の基盤技術と関連研究
- c) 仮説推論の実用エキスパートシステムへの適用法
- d) 仮説推論を発展させた次世代知識ベース・ソフトウェア・アーキテクチャ

3. 活動内容と成果

上記の4項目の分類に従って、活動内容と成果の概要を記す。

a) TMS, ATMSを中心とする実用技術としての仮説推論とその高速化

仮説推論のうちでAIツールに取り入れられたり、応用が進められているATMSと、その起源であるTMSについてまず実状の把握を行った。これらの扱うデータ/知識は変数を含まない命題(論理式)の範囲だが、推論の高速化に研究開発が傾注されている。

J. Doyle によるTMS (Truth Maintenance System) は、Problem Solverから与えられる justification に基づいて、仮説、ノード間の真偽関係を維持する。TMSはTMアルゴリズム、DDBアルゴリズムを基本的機能として動作する。TMアルゴリズムは、あるノードの状態(in, out)が変化したときに、ノードの状態の変化の伝播を行い、各ノードの状態を再計算するアルゴリズムである。ノード間の関係がloopを構成する場合にも対処するため、アルゴリズムが複雑になっている。DDBアルゴリズムは、矛盾が生じた場合に矛盾を解消するアルゴリズムであり、矛盾に関係したノードの状態を変化させることにより、矛盾を宣言したノードをoutにするものである。(紹介した久野委員(東工大)はTMSを物語理解に応用した。)

逐次型推論による推論速度の問題の他に、ノードレベルの状態の整合を行うがjustificationレベルの評価に不向きである点、バックトラックの際にどの仮説をoutにするかの決定が十分考慮されておらず、foundationのルートから行き当たりばったりの仮説の操作を行っている点等の問題が挙げられた。

J. deKleer によるATMS (Assumption-based TMS) はTMSのバックトラックを伴う逐次型の推論を横型の並列探索を行う推論に変えることにより、速度の向上を図ったものである。複数の仮説の選択の可能性がある場合に、仮説の集合を複数の世界(multiple contexts)で実現し、世界内のデータの整合性を維持する。仮説の世界を束(lattice)に展開し、矛盾条件(Nogood)を表わす知識による仮説世界の刈込みを集合演算で行うことにより、バックトラックを伴わない並列推論を実現している。(紹介者の飛鳥井委員(キャンノン)及びICOTでATMSのインプリメントを進めている。)

ARTによるViewpointもATMSと類似の機能を実現している。しかしARTではデータの有効範囲をデータが生成されたViewpointと消去されたViewpointのペアによって表現しており、データのラベルを仮説の組合せで一意に表現するATMSとは異なる。このためARTは仮説の組合せに制約が強い問題に適し、時制推論等に向くが、解空間が複数の仮説の組合せとして表現可能な問題には不向きであり、ATMSの方が適している。KEEにおけるワールドもATMSと類似の機能と考えられる。しかしデータに仮説と事実の区別がない、ワールドが仮説の組合せとして一意に表現されないなどの点でATMSと異なっている。

ATMSでは推論の高速化が図られているが、まだ十分ではない面がある。外部から講演をお願いした飯島氏(富士通研)の「仮説ネットワークを用いた仮説推論器」はProblem Solver側とATMS側の推論ネットワークを融合することにより、推論の効率化を図

っている。具体的にはProblem Solverとして用いているプロダクションシステムの高速照合法であるRET E マッチのネットワークをATMSの並列推論と融合させた高速推論機構を実現している。同様な工夫は飛鳥井委員も行っているとのことであった。

ATMSの仮説推論はいろいろな応用に有効だとの認識はあるものの、更に速度の向上が必要とされている。

b) 論理の枠組上での仮説推論の基盤技術と関連研究

ICOTの特徴を生かすためにも、論理の枠組み上で仮説推論を構成し発展させる必要があり、関連研究のサーベイ、委員自身が進めている研究を基に討議した。ATMSは高速化の点では有効な技術であるが、そのままでは変数を含まない知識である命題論理の範囲しか扱えない。知識表現の範囲を広げるためには変数を含む知識を扱う1階述語論理、あるいはそのサブセットであるホーン節論理(Prolog)上に構築を図る必要がある。

1階述語論理に基づく仮説推論についてはD. L. Poole, R. Gobel (ワートルロー大)らによる先行研究がある。これは複数の仮説の集合が生成されたときには質問応答によって仮説を一個に絞り込む機構を含み、故障の診断等に直接的に応用可能である。論理に基づく仮説推論が設計問題にも適用可能であることを示唆した研究にRESIDUE (スタンフォード大)がある。

石塚委員(東大)は、論理に基づく仮説推論を質問応答を通じて原因を絞り込む故障診断に適用しやすいように知識表現を拡張したシステムについて紹介した。このシステム上で与えた例からの仮説を含む知識の帰納推論を行う概念学習機構を開発し、矛盾を含む知識も仮説としてならば学習できることを示した。更に、仮説を含むフレーム型知識ベースに対する知識同化、調整機構について示した。ここでは仮説推論の非単調性が考慮されており、また矛盾発生時に矛盾知識を同定するためにShapiroのデバッグ・アルゴリズムを応用した質問応答機能が含まれている。

推論速度については仮説に重みを設定し、重みの少ない仮説の集合からの優先探索を導入したが、速度の向上は依然として大きな課題である。変数を含む述語論理式でもエルプラン領域に展開すれば原理的にATMSのような集合演算による並列推論が適用できることが議論されたが、これは膨大な数となり実用的でないことから、部分的な展開手法等が必要と考えられる。石塚等のシステムは論理上では学習機能などを実現するため、Prolog上でMESON証明手続きによるFullの1階述語論理上に実現しているが、これはPと $\neg$ Pの同時存在の矛盾性を常にチェックしていることになり、速度上は好ましくない。矛盾をチェックする制約知識は別に管理する方が実用上は好ましいことが議論された。

井上氏(ICOT)は、R. ReiterとJ. deKleerによるCMS(Clause Maintenance System)について紹介した。CMSはATMSを次のように拡張している。(1) ReasonerがCMSに与えるjustificationはHorn節だけでなく、任意の節であってもよく、結論はアトミックな節でなくてもよい、(2) CMSに対してReasonerは任意の節を質問でき、これに応じてCMSはその節に対するminimal support(アトミックな節の組)を返す。論理と実用的なATMSの関係を明確にした点、欠落している知識を生成するAbductionである点、主項展開により知識ベースをコンパイルすることによる高速化手法を示している点で、今後新機能を考える上で、参考になるシステムである。

中川委員（横浜国大）はサーカムスクリプションに関連して研究している階層的知識の再構成に関して紹介した。サーカムスクリプションは実用的には他の代替手法の方が容易であることが多いが、不完全な知識を扱う上での理論的基礎となる。フレームや知識継承といった概念の理論的裏付けを得るためにこれらを1階述語論理に変換して表現することを考える。このとき

(i) meaningful non-recursive definite clauseの部分クラス

(ii) non-recursive stratified

の2つのクラスがサーカムスクリプションによって不変であるという重要な性質をもっている。多くの論理式はこれらのクラスに含まれるのではないかとのことである。サーカムスクリプションをpartial でやるのは、全体では知識の量が膨大なためとのことである。

桑原氏（オブザーバ、NTT）からP. Delgrande の'An Approach to Default Reasoning Based on a First-order Conditional Logic' の論文紹介があった。この1階条件論理では1階述語論理に' $\Rightarrow$ 'というオペレータを導入してデフォルト知識を記述し、可能世界意味論に基づいて真偽を与えるようにしている。 α が真であるような例外の最も少ない世界で β も真のとき、デフォルト $\alpha \Rightarrow \beta$ は真であるとしている。（詳細は原論文やSWG資料を参照ください。）Reiterのデフォルト論理は人間には分かり易い表現を与えているが、proverや解釈の論理的基盤が必ずしも堅固でなかったが、この1階条件論理はそのような一つの論理的基盤を与えていると考えられる。論理に基づく仮説推論を考える論理的基盤としても関係が深いと思われる。

中川委員からはD. Pooleの'Variables in Hypotheses'の論文紹介があった。Pooleらの論理に基づく仮説推論では与えられた観測事象を説明するために変数を含む論理式の集合である仮説素からground instanceを仮説として生成しているが、仮説をground instanceに限ることが好ましくない例題を挙げ、スコレム定数を定義した文脈を無視できないことを示している。仮説における変数は関数を含まない述語論理なら問題ないらしいが、関数を使わないと存在記号がどう処理されているか分らないし、導出原理も使えない。そこで関数を含まない述語論理を扱うには、スコレム関数が出してきた素性という意味で考慮を要するというのが、この論文の意義である。述語論理に基づく仮説推論を高速化するためにエルブラン領域にインスタンス化してATMSの機構を適用することが考えられるが、その際に考慮すべき一つの観点となろう。

久野委員からは、J. Pearl の'Distributed Revision of Composite Beliefs' の論文紹介が行われた。非単調推論のBelief Updating & Revision of Belief Commitmentsのモデルを紹介し、それを分散アルゴリズムで行う方法を示している。事象をベイズ的確率が付されたノードとし、因果知識をノード間をつなぐリンクとして表わし、ネットワーク表現を行い、この整合をとる方法となっている。確率的要素を持つ環境で並列分散的に信念の更新や翻意を行う方法である。デジタル回路診断や病気の診断の例が示されている。隣接するノード間のローカルな計算から、与えられた入力を最も良く説明する仮説の集合で抽出できるところが面白いとの指摘があった。

仮説を広くとらえると不完全な知識ということになる。Sowa (IBM) は、不完全な知識を含む知識ベースを知識スープと名付け、その課題を挙げていることが井上氏 (ICOT)

により披露された。不完全な知識を論理的基礎の上で扱う場合、非単調推論が必要となる。これまでに各種の非単調推論や非単調論理が個別に提唱されてきたが、1987年はそれらの相互関係、共通的な基盤が次第に明らかにされた年であるといえる。このような関係の把握はある仮説推論の枠組みを選択するに際しても、その位置づけと適用範囲を知る上で重要なことである。

c) 仮説推論の実用エキスパートシステムへの適用法

仮説推論は現実の問題の記述、モデル化にも有効だと考えられるが、まだ本格的応用システムの例は少ない。仮説推論の実問題への適用法、必要な機能を明らかにするため、委員によるシステムにおける考え方、文献調査による検討を行った。

丸山委員（富士通研）からはVLSI設計エキスパートシステムの紹介があった。これは動作記述、構造記述、制約条件を入力とし、CMOS回路を出力する。動作記述（動作アルゴリズム）から制御回路設計を行うエージェント、構造記述（データパス）からスタティック回路設計を行うエージェントが制約条件を満足するように協調的に動作する構成である。ある機能を持つ回路を構成する場合、その回路構成には複数の実現方法があるが、このような選択肢がある場合には、これらの選択のそれぞれを仮説と考える。推論過程は、理由付け（Justification）として記録される。矛盾が生じた場合には、その矛盾を導出した仮説の集合をJustificationをたどることにより求め、その中のひとつの仮説を選択肢の別の仮説に変更することにより、矛盾の解消を行う。同時にJustificationをたどり、Dependency-directed Backtrackingを行うことにより、Truth Maintenance（変更された古い仮説から導出されたデータの消去）を行う。上記の仮説の選択はドメイン知識を用いて行う。DoyleのTMSの観点から解釈すると、評価値の制約条件違反により矛盾の検知を行い、矛盾を解消するための仮説の変更は、Justificationをたどることにより、基本的にはドメイン知識を用いて行われる。仮説推論を実際の問題に適用する場合の、仮説や矛盾などについて次のように考え方を明確にしている非常に興味深い。

仮説……選択肢における選択

TMSのようにあるデータがINでないときにINになるデータ、ATMSのように自分自身を理由付けに持つデータではない。

矛盾……制約条件の違反

TMSやATMSのように仮説の組合せではなく、Justificationがnogood justificationとして記録される。

矛盾を論理的矛盾（ $A \& \sim A$ ）のように、捉えるのではなく、制約条件の違反と捉えることにより、矛盾の検出の効率が良くなっている。

永井氏（ICOT）からはBoseとPadalaの'Reasoning with Incomplete knowledge in an Interactive Personal Flight Planning Assistant'に記された対話的に個人のフライトプランニングを行うための支援システムにおける、不完全な知識を用いた仮説推論について紹介があった。フライトプランニングのような問題は、制約条件を満足する問題に分類することができ、この問題の場合の制約条件は、出発地、目的地、日時、コストなどであり、互いに依存関係にある。制約条件のすべてをあらかじめ与えることは困難であるため、いくつかの制約条件は仮説としてデフォルト的に用いて推論を進めることになる。こ

の推論の過程は、justification として、データが導出されたデータの集合により記録される。また、あらかじめ与えられた制約条件もユーザとの対話の過程で変更されることもあり、このようなときに、Truth Maintenance が行われる。システムは大きく分けてフライトスケジュールのリレーショナルデータベースと、問題解決とTruth Maintenance を行うARMS (Assumption-based Reason Maintenance System) から構成され、ARMSでの仮説推論により得られた結果がdatabase queryとしてデータベースに送られる。本システムでは、ユーザが入力した制約条件により、仮定されたユーザのモデルやそのモデルから導かれる制約条件が仮説となり、それらの仮説の集合が、現在のユーザの信念空間 (Belief Space : active-context) を表現する。新たな制約条件の追加により、矛盾が生じた場合には、仮説を変更することにより、信念空間の変更が行われる。矛盾の解消のためのTruth Maintenance は、ATMSに近いものが用いられているらしい。この論文や、丸山氏の研究紹介では、制約条件の緩和のモデル化が重要な問題である。この問題はNP-hardな問題であるがユーザのモデルを導入し、重要視する制約条件の項目を限定することにより対処し、実用に供している。

大貝氏 (オブザーバ、新日鉄) は、熱延加熱炉 (複数) からの抽出順番問題にARTのView-pointによる仮説推論を適用した。知識表現簡便で非常に使い易く、ルール数も小であることは評価が高かったが、速度が大きな問題であることの指摘があった。この問題はVAX780上のARTのViewpointで40分もかかった。同じ問題を μ VAX上のOPS5で再構築したところ、ルール数は増大し、メンテナンス性も良くなったが、問題解決時間は40秒と向上したとのことである。

赤埴氏 (オブザーバ、NTT) からはモデルに基づく診断について仮説推論との関係で考え方の紹介があった。診断対象の構造や動作などを記述したモデルが与えられた時に、モデルから予測される挙動と実際に診断対象で観測された挙動との相違点からその原因 (故障箇所) を推論する診断方式についてである。関連研究として、deKleerのGDE (General Diagnostic Engine) は多重故障を対象としATMSを利用して故障の探索を行い、また経験的知識 (各構成要素の故障する確率) を融合するため、情報理論 (最少エントロピー法) を導入している。また、Reiterの論文はモデルに基づく診断の定式化と診断を計算する論理的アルゴリズムについて述べている。

和田氏 (NEC、森委員の代理) は電子交換機の故障診断システムを開発中である。不確かなことを仮定し、それに基づいて推論を行い、実行中に誤りや不都合が見出された場合には仮定や仮説を取り消して推論を切り替え、別の筋道をたどる推論の適用について紹介があった。実用的な不完全な知識やデータの取り扱い手法となっている不確実性の数値を矛盾の検出に応じて変化させる考え方などが示された。

今村委員 (機械技研) からは機械設計、具体的には旋盤設計において、非決定的選択を行うことが多く、効率的な探索、無矛盾性の管理の面での仮説推論への期待が述べられた。設計における仮説生成、選択、検証は仮説推論と関係が深い。

加藤委員 (松下電器) からも工作機械の主軸設計問題を例にとり、仮説推論への期待と現状の問題点が述べられた。無矛盾性の管理、制約条件を緩和する制御、Multiple Contextsの管理などの点に関してである。

A T M S も含む仮説推論は応用面での期待は大きいものの、実用とするためには速度の面を始めとして解決しなければならない課題も多い。特に変数を含む論理の枠組み上では速度の問題の解決がせまられている。

d) 次世代知識ベース・ソフトウェア・アーキテクチャの基盤技術としての仮説推論

仮説はより広くとらえると不完全な知識の一種と見なすことができる。石塚委員による不完全な知識とはより平明に述べると、常に成り立つ（真である）とは限らない知識であり、より具体的には、例外を含む知識、欠落部分のある知識、矛盾の可能性のある知識、仮説的な知識、拡大解釈の知識、変る可能性のある知識、などである。完全な知識に加えて、これらの不完全な知識を含めて知識ベース（知識スープ）を構成し、それを操作することによって演繹機能を超える高次人工知能機能を発揮する次世代知識ベースの姿が議論された。完全な知識には制約的知識を含む。知識の拡大解釈（更に具体的には定数項を変数項へ読み換えること）は特に重要な基盤的知識操作技術である。これは類推、常識、帰納推論、発想などの高次人工知能機能とも結びつく。従って、このような高次人工知能機能にアプローチするには、知識の変数化表現は必須となる。仮説推論を実用的にも重要な出発点とし、高次人工知能機能をサポートする次世代知識ベースに向けてのアプローチについて議論した。

個々の高次人工知能機能（非単調論理、類推、帰納推論、学習、発想など）を追求するのも勿論重要であるが、次世代知識ベース・ソフトウェア・アーキテクチャの全体の見通しを明らかにし、個々の研究を位置づけることが必要と思われる。仮説推論を出発点とし、S W G の活動を通じてそのようなソフトウェア・アーキテクチャのおぼろげな姿を示したつもりである。

4. まとめ

仮説推論に関係する次のような種々の課題について議論を深める今後の方向を討議することができた。

- a) T M S、A T M S を中心とする実用技術としての仮説推論とその高速化
- b) 論理の枠組み上での仮説推論の基盤技術としての関連研究
- c) 仮説推論の実用エキスパートシステムへの適用法
- d) 仮説推論を発展させた次世代知識ベース・ソフトウェア・アーキテクチャ

I C O T ならびにその他機関における今後の知識システムをサポートする次世代ツールの研究開発に役立つことを期待したい。

2. 次世代知識ベース・ソフトウェア・ アーキテクチャへのアプローチ

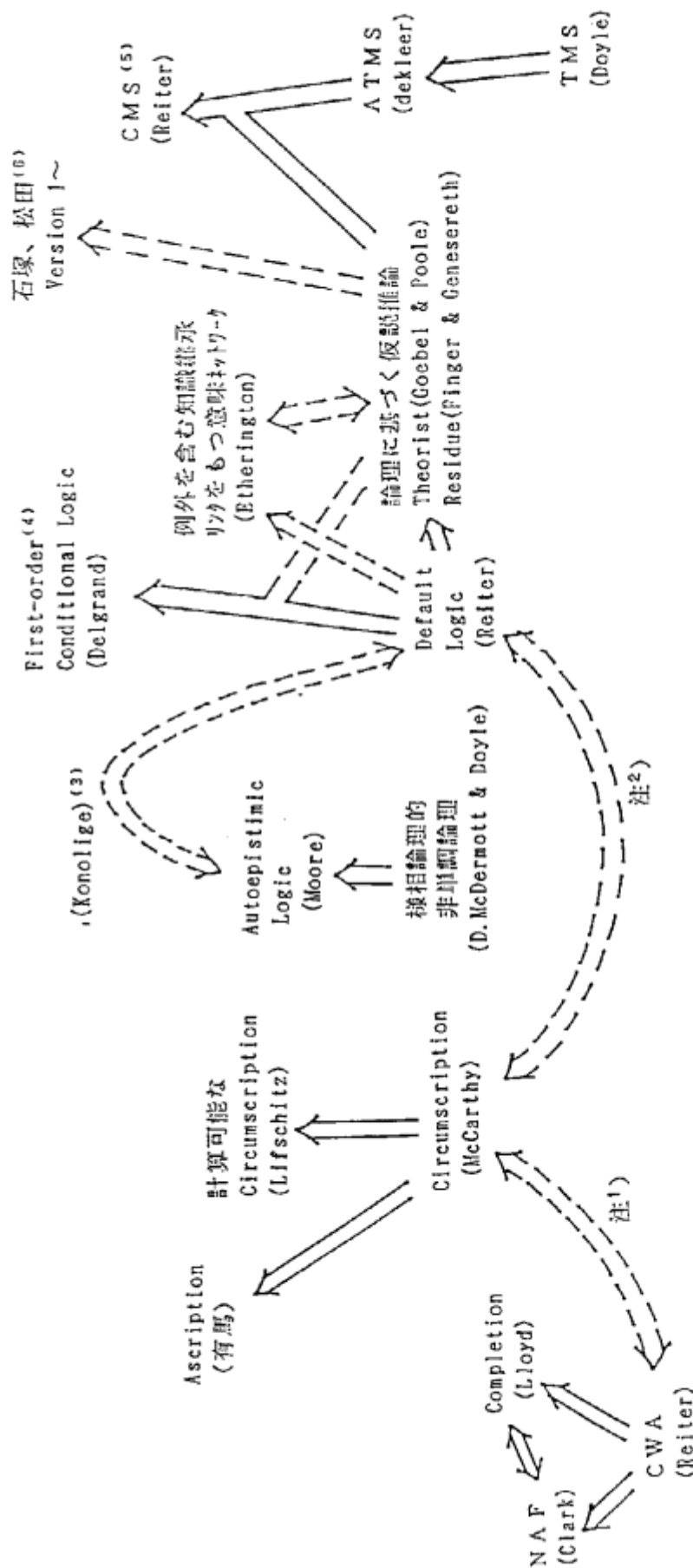
石塚 満
(東京大学 生産技術研究所)

我々は「完全な知識に加えて不完全な知識も含めて知識ベースを構成し、これを操作する高次推論によって高次人工知能機能を実現する」ことを基盤技術として次世代知識ベース・アーキテクチャを構築することを目指している。不完全な知識も含む次世代知識ベースへの出発点として論理に基づく仮説推論を選んだ訳であるが、この選択は今日でも正しかったと考えている。

不完全な知識の表現と操作に関しては、仮説推論以外にも種々の手法が考案されている。1987年はこれら種々の手法間の関係が次第に明らかにされてきた年であったといえる。図1はこれら種々の手法とその間の関係を表示している。

CWA (Closed World Assumption、閉世界仮説)、NAF (Negation as Failure)、Completion は不完全な知識の扱いに関する今日では古典的な位置にある手法といえる。CWAは記述してあることのみが真で、記述されていないことは存在しない、あるいは偽とする考え方であり、広く取り入れられている。NAFは証明できなかったとき偽とする考え方であり、Prolog の not は論理否定とは異なるこの考え方(証明論的な not)によって実現されている。Completion は $\sim$ if ... を $\sim$ if and only if ... として解釈する考え方である。

不完全な知識が含まれると推論が非単調的になる。非単調推論には Reiter のデフォルト論理、D. McDermott & Doyle の様相論理的非単調論理、McCarthy のサーカムスクリプション(限定するといった意味)の流れがある。サーカムスクリプションは記述してあることのみが存在するといった閉世界仮説(CWA)に近い考え方であり、厳密な論理的扱いが展開されている。最近では McCarthy の弟子の Lifschitz により、計算可能性を考慮した方向へ展開されている。厳密な論理的扱いがサーカムスクリプションの特徴だが、実用性の観点か



注¹) 確定節のみときは一致。一般の節のときは、

Circumscriptionの方が強い。階層化(Stratified)データベースを最小モデルを求める順序とすると

Prioritized Circumscriptionと一致(Lifschitz)

註²) Imielinski (Artif. Intell. Vol.32, 1987)

Etherington (IJCAI-87)

JCAI-87

注4) Default Logic の Prover は仮説推論の方法で実現できる (AAAI-87)

注⁵) AAAI-87

注⁶）別紙に發展型を示す図

不完全の知識に関する諸手法と関係

らは他の代替的手段によっても対処できるので、学問的意味の側面が強い研究の流れといえる。

様相論理的非単調論理は様相的扱いを導入して不完全な知識に関連する非単調推論を構成したものだが、不動点の定め方などの課題が伴っていた。Moore の Autoepistemic Logic (自己認識論理) は、信じる (belief) ことのみを正しいとして考える方法で、様相論理的非単調論理の課題を解消し、やや実用的にした流れと位置づけられる。

デフォルト論理は“鳥であって他の知識と矛盾することがなければ飛ぶとする”のように、人間にとって自然な一つの不完全な知識の表現法を与えている。性質の継承の階層構造をもつ意味ネットワークにこの考え方を入れたものが、例外を含む知識継承リンクをもつ意味ネットワークである。論理に基づく仮説推論も、意味内容的にはデフォルト論理と近い関係にある。デフォルト論理は人間にとって自然な一つの表現法を与えているのだが、その prover (証明器) は完全性、健全性などの点で必ずしも十分に規定されているものではなかった。

Delgrand の First-order Conditional Logic (1階条件論理) はデフォルト論理の流れを論理の上に展開し、論理的な prover を与えたものとして位置づけられる。サーカムスクリプションとデフォルト論理の関係も示されている。最近では、Autoepistemic Logic とデフォルト論理の関係も議論が進んでいる。

仮説的な不完全な知識の扱いに関する現時点で最も実用的な手法は、A T M S (Assumption-based T M S) であるといえる。A T M S では横型の並列探索で集合演算的に仮説を絞り込む機構により、高速化が図られている。A T M S は論理的枠組みとは離れて議論されてきたが、最近 Reiter は論理に基づく仮説推論に近い枠組みで、A T M S を一般化した C M S (Clause Maintenance System) を発表している。A T M S にしても C M S にしても現状では変数を含まない命題論理の範囲にとどまっていることが、知識の表現の能力、特に今後の知識の拡大解釈などを行う上で、大きな制約であると思われる。

以上、不完全な知識の表現と推論に関する諸手法について述べてきたが、図1に表示されているように次第にそれらの間の関係が明らかにされつつある。いずれ研究が進展すれば不完全な知識に関する統一的な枠組み、手法が確立されるかもしれないが、現時点で論理に基づく仮説推論を次世代知識ベースへの出発点に選択する理由は、主として次の2点である。

- 1) 1階述語論理を背景とした枠組みをもち、発展性があること。
- 2) 診断問題や設計問題などへ比較的直接的に適用できるなど、実用的な有効性をもつこと。

我々が論理に基づく仮説推論を出発点に選択したのは3年程前であったが、この選択は今日でも正しいと考えている。

我々は論理に基づく仮説推論を出発点にして、実用的な成果を生み出しながら次世代知識ベースへのアプローチを図2のように設定している。実用的観点からまず必要なのが推論の高速化である。我々は既に仮説に重み（ランク）を付し、重みの和の少ない仮説集合を優先して生成、検証する最良優先探索法を組み込んだ仮説推論システムをインプリメントした。故障診断の場合は、よく起こり易い1重故障から優先して調べていくことになる。5ゲート程のデジタル回路の故障診断では約10倍速度が得られる。更に高速化のためにATMSの並列推論による仮説の絞り込みメカニズムの導入を図るべく検討を始めている。原理的には変数を含む1階述語論理の知識でも（関数はとりあえずなしとする）、エルブラン領域に展開すればATMSのメカニズムを適用できる訳であるが、これは量的拡大が非常に大きく実用的でない。そこで、部分に分割してエルブラン領域に展開する方法等をとる必要があると考えている。

仮説知識を含む知識ベースへの知識の同化、既存知識を無矛盾性を保つように調整を行う管理機構については、既に我々が開発している。高速化技術と知識ベース管理機構を組み込んだ仮説推論システムに、充実したユーザインタフェースを付加すれば診断問題用のツールとなる。このとき、症状と原因を結ぶ因果関係を一つ一つ記述しなくてもよい点が、従来のツールとの大きな違いであり、知識獲得フェーズでの大きな利点となる。

その先の発展は二つの流れを想定している。第一は制約的知識の利用であり、第二は類推機能の付加である。

設計問題では探索空間を狭めるために制約を有効に使用することが重要になる。仮説推論における矛盾チェック用の知識も一種の制約、更には外部から与える観測も一種の制約と見なすこともできる。仮説生成のとき使用する能動的制約、仮説検証時に使用する受動的制約の効果的利用法の開発が必要と考えている。更に、適切な仮説が生成されないとき、一部の制約を取りはずす Relaxation（弛緩）の機能が重要となる。

制約知識の利用を含む仮説推論に充実したユーザインタフェースを付加することによって、設計用のツールとすることができる。知識型設計に対しては方法論、パラダイムがまだ確立していないが、制約知識操作を含む仮説推論はその有力候補になると考えている。

類推は知識を拡大解釈（定数項の変数化解釈など）に基づいて実現する。この操作は例からの帰納推論で用いた技術と類似なものである。類推は丁度使える適切な知識が存在しない

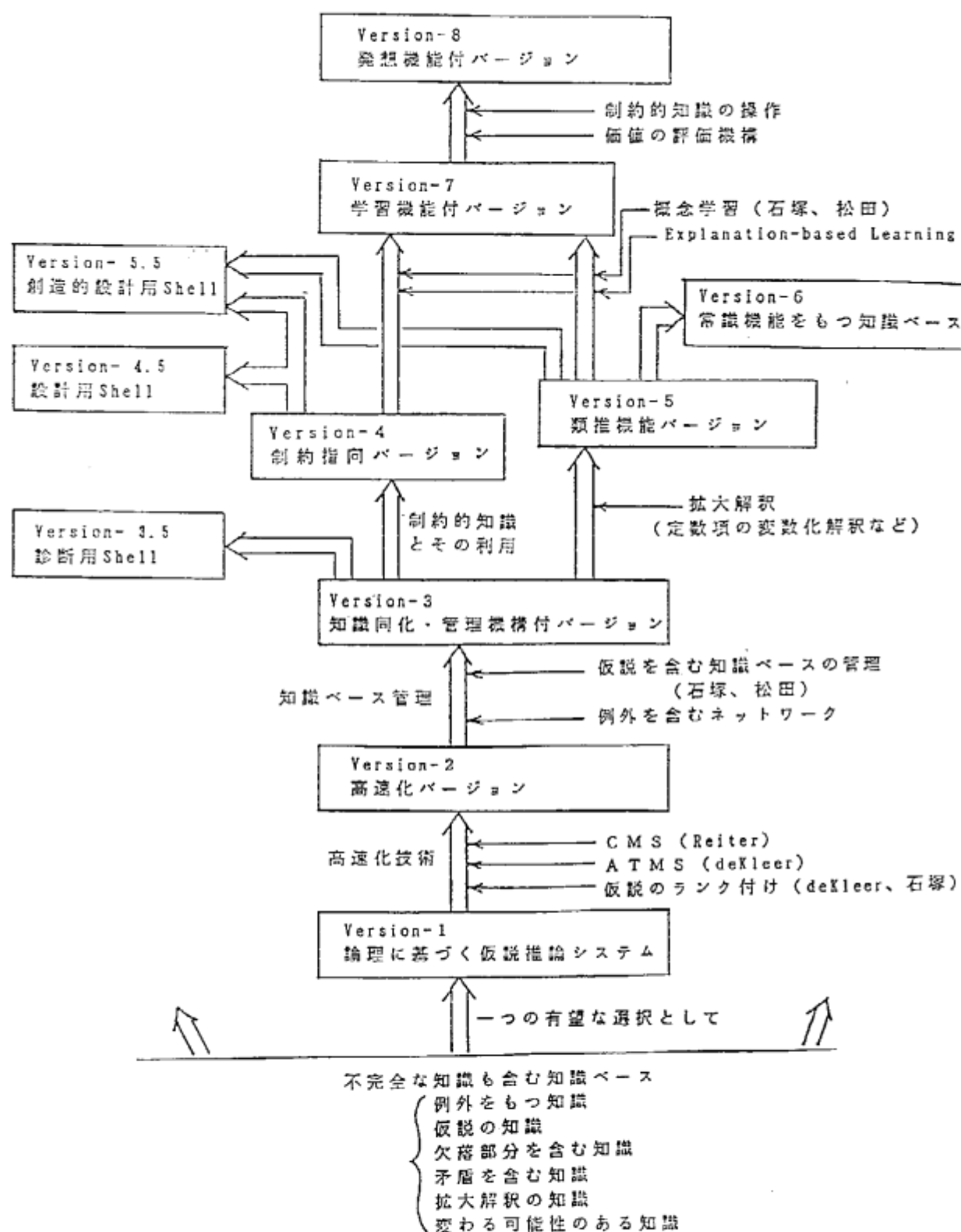


図2 次世代知識ベースへのアプローチ / 7

場合に、類似な知識を拡大解釈して代用するように働かせる。一般にどの知識に対して拡大解釈を働かせるかが問題となるが、CMSの欠落している知識を見出すメカニズムを利用して類推を働かせて対象知識を限定することを検討している。

類推機能バージョンに基づいて実現する大きなシステムは常識機能をもつ知識ベースである。現在の知識ベースは知っていることには答えられるが、少しでも知らないことには対処できないといった福の狭さ、脆弱性が大きな問題となっている。常識機能はこの問題の解決への道である。MCCのLenatも類推に基づく常識機能を備えた大規模知識ベースの開発を進めているが、彼らのアプローチは論理の枠組みとは離れている。ここに示すように、論理を基盤として不完全な知識の操作の一環として類推、そして常識機能の実現を図るアプローチの方が大きな広がりが見込めると考える。

類推のもう一つの適用領域は創造的設計である。単なる組合せ的新しさだけでなく、既存の要素部品パターンから類推し、仕様を満たすような新たな要素部品、組合せを生成することが期待できよう。

仮説を含む知識の例からの帰納推論法を示したが、実はこのような学習機能が役立つのはやや先のことになると思定される。その理由は、今日の帰納推論は例から初期的知識（ルール数で20-50程度）を生成するのには使えるが、その機械によって生成された知識は必ずしも人間に分かり良い形になっていない問題がある。本当に機械学習が必要とされる局面は、例えば既に1000のルールが存在し、あとどのようなルールを加えたら性能が向上するのか人間では判定できないような場合であると考えられる。そのような局面に対応できる技術の研究開発が今後の必要と考えている。複数例からの帰納推論（最近ではSimilarity-based Generalizationとも呼ばれる）の他に、説明に基づく一般化学習（Explanation-based Generalization）も一定の役割を果たしてくるであろう。

発想についてはまだ話だけの段階であるが、一応次のように考えている。先に制約的知識を一部取り除く操作と知識を拡大解釈する操作について述べたが、この二つの機能を利用する。即ち、制約的知識を一部取り除き、もてる知識を拡大解釈し、発散的におもしろく、価値ある組合せを生成する。そして、この組合せに合うように逆に制約的知識を組み換える。ここで難しいのが価値の評価機構であり、多分に右脳的能力との関係が深くなってくる。Lenatは発見を行うシステムAMやEURISKOを作成したが、多分にAd hocな技術を駆使して作成したので後続のシステムが現れなかった感じがする。ここに示すように仮説推論を基盤として、発想のメカニズムまで組み上げれば、波及効果は極めて大である。

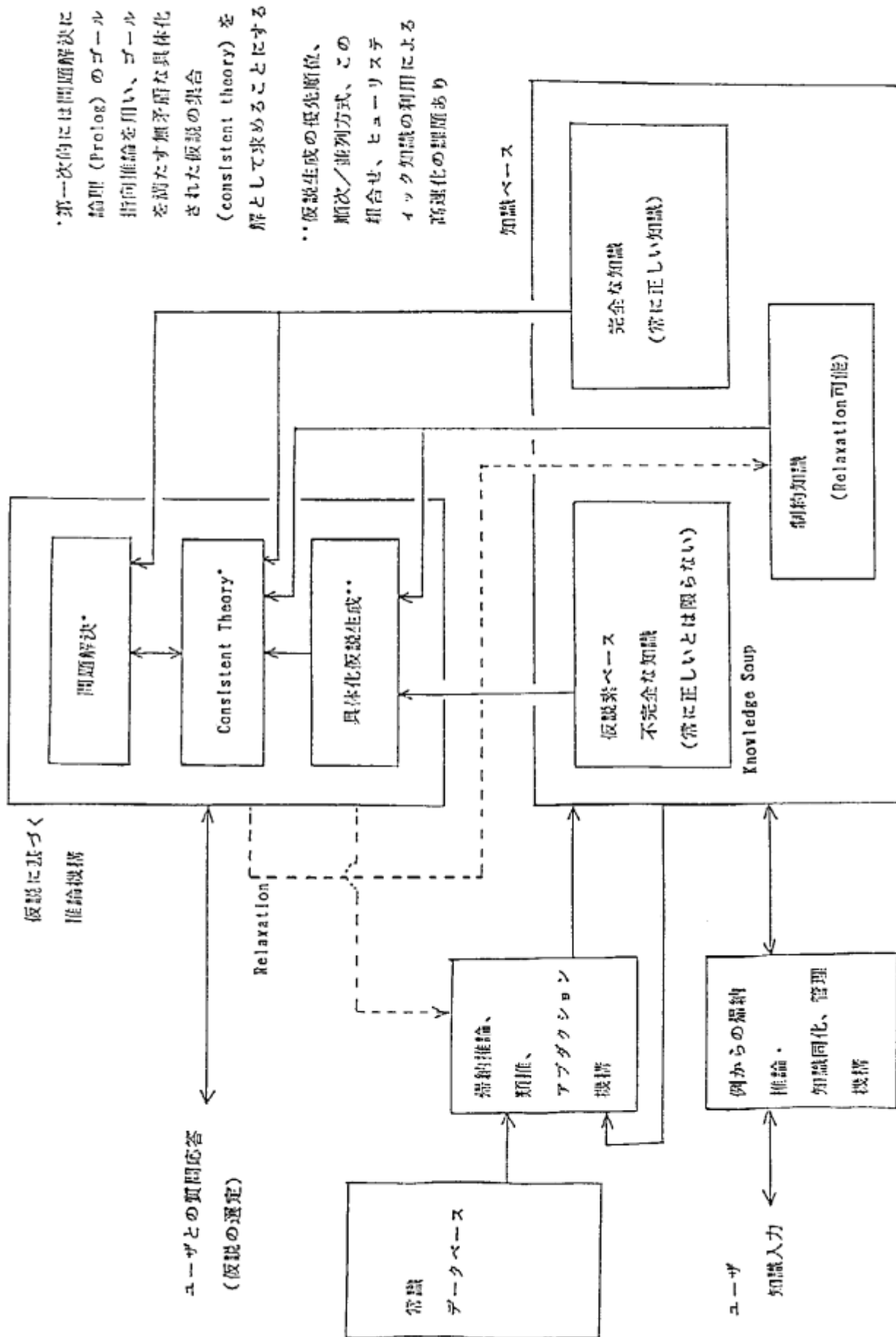
以上、我々が考えている不完全な知識の操作を基幹とする次世代知識ベース・ソフトウェア・アーキテクチャへのアプローチを示した。このような全体像の枠組みの上に基盤技術、実用技術の整備を進める必要性を主張し、今後とも継続して研究開発を推進する所存である。高速化のためには並列／分散処理との組合せも重要と考えている。

図3は当面の目標としている次世代知識ベース・システムの形態を示している。この基盤技術の研究開発と並行して、実現した高次人工知能機能の利用を図る応用システムの研究開発も進める所存である。

図4は現代のプログラミング、そして現在の第一世代ともいえるべき知識ベース技術、その延長上での次世代知識ベースを分かり易く展望し、それらの構造と機能を示している。

<参考文献>

- 石塚 満：最近のAI技術、テレビジョン学会誌、Vol. 41, No. 9, pp. 772-779 (1987)
- 石塚、松田：不完全な知識環境下での高次推論、富士通国際研夏期シンポジウム「知識システム方法論」pp. 139-159 (1986)
- 石塚 満：曖昧な知識の表現と利用、情報処理、Vol. 26, No. 12, pp. 1481-1486 (1986)
- 松田、石塚：仮説推論システムの拡張知識表現と概念学習機構、人工知能学会誌、Vol. 3, No. 1, pp. 94-102 (1988)
- 松田、石塚：仮説知識を含むフレーム型知識ベースでの知識同化・管理機構、電子情報通信学会論文誌(D)、採録決定
- 石塚 満：画像の認知と仮説推論、電子情報通信学会、パターン認知と画像理解研、PRU87-123、及び画像工学研、IE87-137 (1988.3)
- 青山、石塚：仮説の重み利用による高速化機構を有する仮説推論システム、電子通信学会全大、D-359 (1988.3)



・第一次的には問題解決に
論理 (Prolog) のゴール
指向推論を用い、ゴール
を満たす無矛盾な具体化
された仮説の集合
(consistent theory) を
解として求めることにする

・・仮説生成の優先順位、
順次/並列方式、この
組合せ、ヒューリステ
ィック知識の利用による
高速化の問題あり

図 3 次世代知識ベース・システムの形態

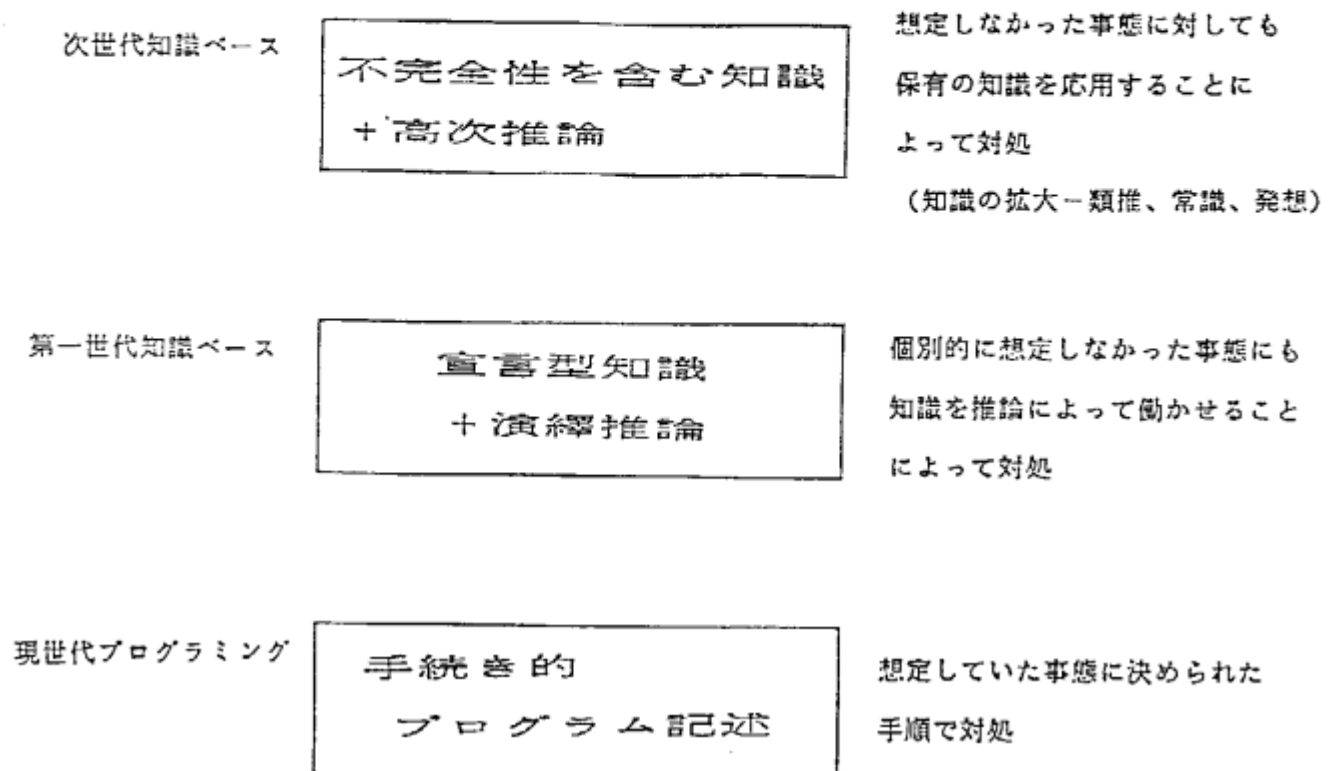


図 4 対応の幅の広がり

3. 仮説推論を利用する枠組み

中川 裕志
(横浜国立大学 工学部電子情報学科)

仮説推論とは何か、技術的にどのような発展が期待できるのか、どんな分野に適用できるのか、... について今までの議論では必ずしも明確な答えが得られたとはいえないが、それなりに問題点も明らかになってきていると思う。これらについて、きちんとした答えを出すためには、境界条件すなわち仮説推論を利用する枠組みを設定する必要があると考える。

仮説推論利用の枠組みのラフスケッチを図に示す。

帰納推論、類推、アブダクション、ascriptionなどの理論が仮説推論に寄与する位置づけが、総合的なシステムでは重要になるであろう。

- (1) まずこれらが、起動される契機となるのは、やはり仮説推論によって得られた理論によって問題解決が計られる（かつ必ずしも成功しない）場合であろう。その場合に必要な知識をなんらかの方法で作り出さなくてはならない。帰納推論などはその場合に起動される。
- (2) 帰納推論等によって得られるのは、個々の（具体化された）仮説そのものではなく、仮説を生成する素になるもの（仮説素）であろう。具体的にいえば、自由変数を含む一般的な規則と考えられる。そして、生成された仮説の無矛盾性も保証されていない。
- (3) 仮説素ベースを領域対応な無矛盾な部分集合に分割するのは、変数を含むルールに対するATMSのようなものを考える必要がある。
- (4) 仮説推論エンジンは、固定化された領域固有の知識ベースに仮説を統合して新しい無矛盾な理論を作る。いわゆる仮説推論の部分である。これそのものが問題解決になっている場合（回路の故障診断等）もある。
- (5) 仮説推論エンジンで使うのは具体化された仮説であり、具体化された仮説を仮説素ベースから切り出す作業は、仮説枚举器による。仮説枚举には、逐次的、並列、両者の組合せ、仮説素の選択の優先順位、等検討すべき問題が多い。

ここ数年の間に進歩した演えき推論以外の推論手法をどのような場面でどのような役割で使うのかを、知的活動全般のなかで位置づけていくことを、そろそろ考えてもよい時期であろう。仮説推論は帰納推論や類推などと協調する技術であると考えられる。

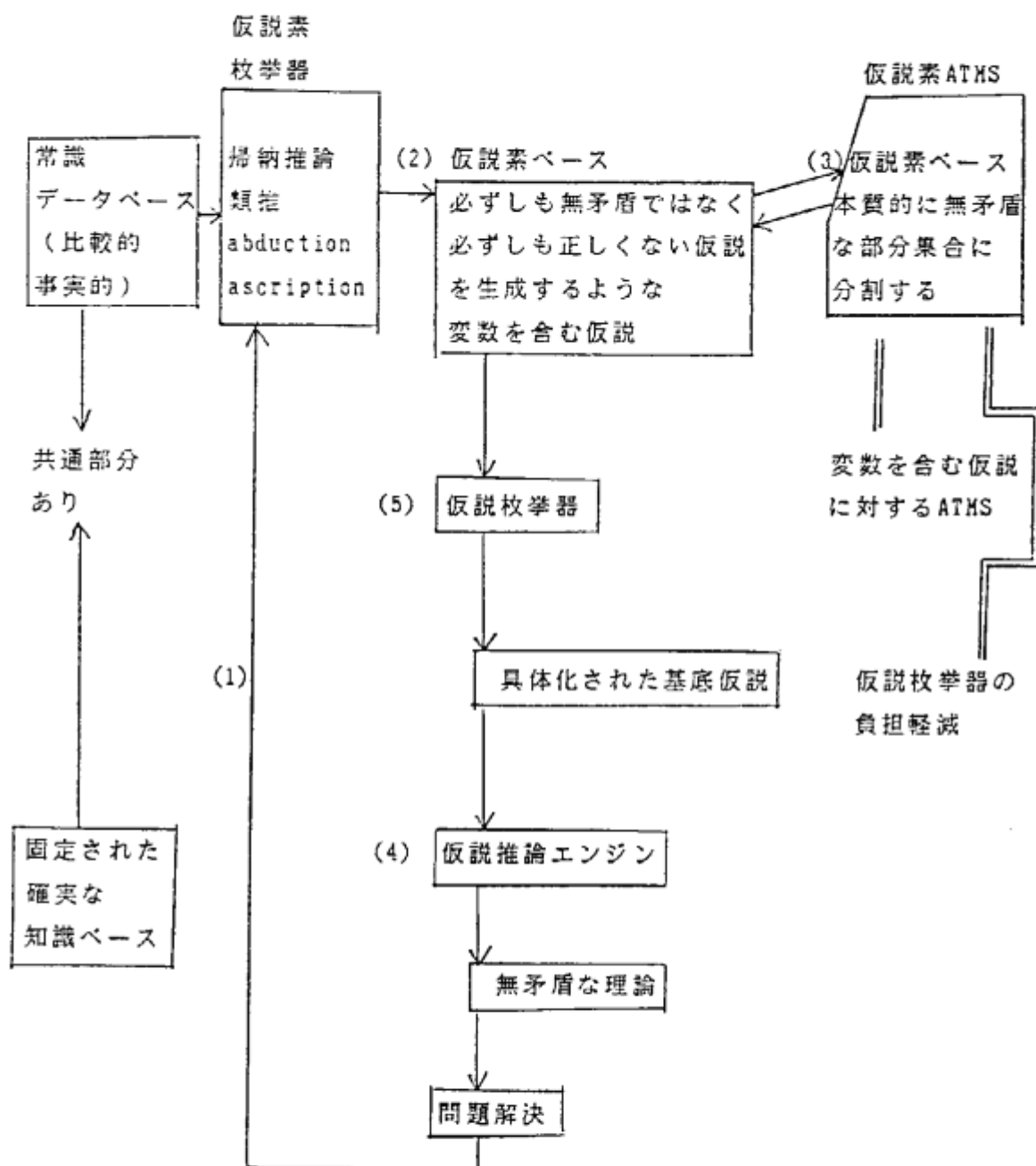


図 仮説推論を利用する枠組み

4. 仮説推論システムのイメージ

久野 禎子

(東京工業大学 理学部情報工学科)

A. “仮説を扱う場を提供する”という次元の問題 (アーキテクチャレベルの問題)

(1) 仮説を扱うデータベースを構築するためには、事象そのものとそれら事象間の関係(dependency)に関する情報を記述する。そのためデータベースはネットワークのイメージとなる。

・事象は観測に基づくもの(given)と観測に基づかないものに大別され、後者のうちデータベースのコンテキストの評価に影響しうるものを仮説という。

・dependency linkは、事象の結合方法を定義するものである。TMSではSL、CPという2通りの方法を用意している。ATMSや論理に基づくシステムでは、linkの意味は特定の範囲の述語論理式として解釈できる。

(2) 次に(1)の情報を操作するアルゴリズムを用意しなければならない。TMSでは、データベースの整合性保持機能(Truth Maintenance)と知識の非単調な増加により生ずる矛盾の解消機能(Dependency directed backtracking)を提供する。ATMSでは、可能なコンテキストすべてを見せるために、与えられたHogood定義のもとでの各ノードのラベルの計算アルゴリズムが中心となる。論理に基づくシステムでは、

既存知識 $\cup$ 仮説 $\vdash$ 観測

を証明する手続きがそのままこれに相当する。

問題 1

(1)の事象の表現に関して、事象中に変数を許すとその扱いが手に負えなくなるという問題点がある。

そこで、例えばそのような事象の解釈に優先順位を付ける等の一般的な戦略を指定できるようにする必要がある。どのような戦略が有効かは実際のところ、試してみるしかないのではないかな？

さらに具体的な応用場面では、各仮説の各変数の意味がかなり決まる場合もあるかも知れない。これには仮説自身が各変数の領域に依存した意味を知っていて、その解釈をコントロールできるような実現になっているのが自然である。

問題 2

(1)のdependency linkの表現と(2)のアルゴリズムに関しては、ユーザの要求によるところが大きいと思われる。

・データベースの中がどのように見えていて欲しいかな？

1つのコンテキストに基づく推論で良いのか、可能なコンテキストがすべて必要なのか。ただし、TMSの一度仮説集合を選ぶと状態遷移ににくいという点と、ATMSの複数コンテキストの並列同時的取り扱いに伴う

大域的情報のアルゴリズムへのしみ出しはトレードオフであり、後者はネットワークの管理をわかりにくくする。

・データベースに対してどのような操作を行なうのか？

・linkの結合方法の指定の自由度は？

TMSではSL, CP以外には？

ATMSでは、choose(or), $\neg$ (not), ignore(preference)等拡張されている。

B. “仮説を用いて推論を行なう”という次元の問題 (仮説推論システムレベルの問題)

(1)システムの全体像(仮説生成・選択・検証サイクルの問題)

問題1—データの内部表現と外部からのマッチングの問題

どのようなシステムを構築する際にも、データをどのような内部表現にし、どのようなマッチングをさせるのかが問題となる。外部からのデータが、変数表現(一般化や特殊化の関係)や領域固有/独立の知識による推論を介して、データベース内で効果的に結合されなければ意味がない。

問題2—知識利用(問題解決)時における仮説世界選択、データベースモニタリングの問題

知識が利用されるときには、適当な仮説の組み合わせを選択して、しかもその下でデータベース全体が矛盾のないことが必要である。

仮説世界選択の手法としては、各仮説世界を評価する何らかの基準が存在してシステム側で調整選択できる場合と、外部から新しい情報を得なくてはならない場合がある。前者の場合、それぞれの応用領域に応じて選択の基準は異なる。

問題3—知識獲得時における仮説修正の問題

獲得途上にある知識は仮説事象として表現され、それに対する関連づけ(すなわち推論)を獲得することが重要となる。新情報が与えられると、既にある知識でそれを説明し取り込もうとするが、その作業は未知の分野については誤っていることがあり修正を要することがある。このようなプロセスでは類推等の知識呼び出し機構が重要となる。帰納や積極的問題解決は、新情報に対しまず仮説を生成して、それを検証・修正するものである。そのような仮説生成・修正は与えられたアーキテクチャ上でいかに実現されるのがよいか？

問題4—問題2、3を通じ、どの範囲の知識(仮説)で選択や獲得を行なうかの問題

知識呼び出し機構の問題。限られた知識総量しか相手にしない応用領域では切実でないが、厳密に扱うにはサーカムスクリプションなどが

必要である。

(2)アーキテクチャとの利用者界面の自由度に対する要求

上述の問題1～4を扱うモジュールの実現に際して、仮説を含むデータベースの構築されるアーキテクチャとの利用者界面にどこまでの自由度が望まれるか決まってくる。たとえば、知識利用・獲得モジュールからデータベースへはどのようなインタラクションが要求されるのか？

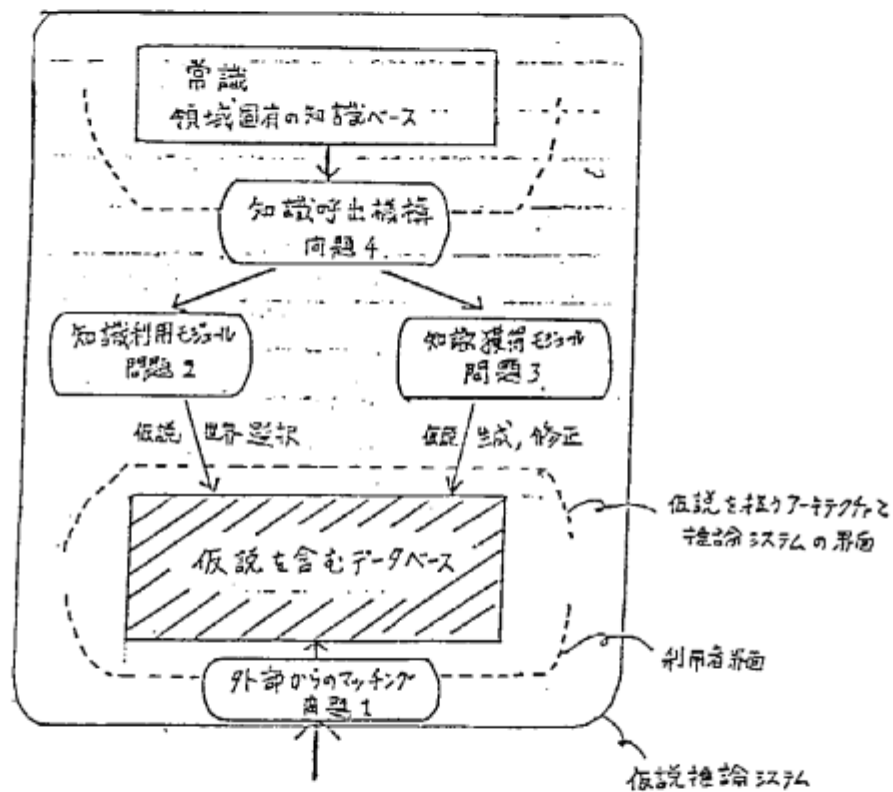


Fig. 仮説推論システムのイメージ

5. 仮説推論を用いた問題解決

井上 克巳
(ICOT 第5研究室)

井上克巳, 「導出を用いた仮説探索」,
情報処理学会第35回全国大会, 論文集, pp. 1567-1568, 1987.

1. 仮説推論とは

・従来のエキスパートシステム (図1): 古典論理に基づいて完全な知識 (真偽が確定している知識) のみを前提とする。

競合する知識や例外を持つような不完全な知識が扱えない。

・仮説推論: 不完全な知識が扱える推論方式の一つ。

真偽が不明である知識や不足しているデータを真と仮定して推論を進める。

制約による無矛盾性のチェックや仮説の再検討, 修正が必要。

一信念の翻意 (belief revision), 非単調推論 (non-monotonic reasoning)。

2. 仮説推論システムの構成 (図2)

・APRICOT (Assumption-based PROblem solver in ICOT):

図1に比べて仮説推論を行うモジュール群 (図2の左半分) を追加。

①仮説生成/列挙: 問題解決にとって必要な仮説集合を求める。

②仮説選択: 仮説集合の中から, 幾つかの仮説の組合せを選択する。

③仮説検証: TMS (ICOT 版) に基づいて仮説の組み合わせの基で矛盾しないデータ集合 (コンテキスト) を管理する。

・TMS (Truth Maintenance System):

推論の実行ごとに結果に対する理由付け (justification) を記録することにより, WMの矛盾性の維持や, belief revisionを可能とする機構。主な例に理由付けに基づくTMS (J. Doyle) と仮説に基づくATMS (J. de Kleer) がある。

・基本姿勢:

①これまで議論されていなかった問題解決の中での仮説推論の役割を明確にする。そのために, ヒューリスティック・ルールを重視して用いる従来の問題解決とは異なり, 問題領域固有の知識や制約知識をも利用できること。

②推論制御を仮説推論と問題解決器の中間に置くことにより, 効率の良い探索が可能な枠組を持つこと。特に, AND/OR木探索に注目する。

3. 仮説推論の応用

①診断: 観測や目標を説明するための無矛盾なコンテキストを目的指向的に仮説を追加しながら形成。

②計画: 種々の制約知識を満足するパラメータ値を仮説にとり無矛盾なコンテキストを形成。

③設計: 階層構造を持つ複数の設計モデルをコンテキストとして管理。

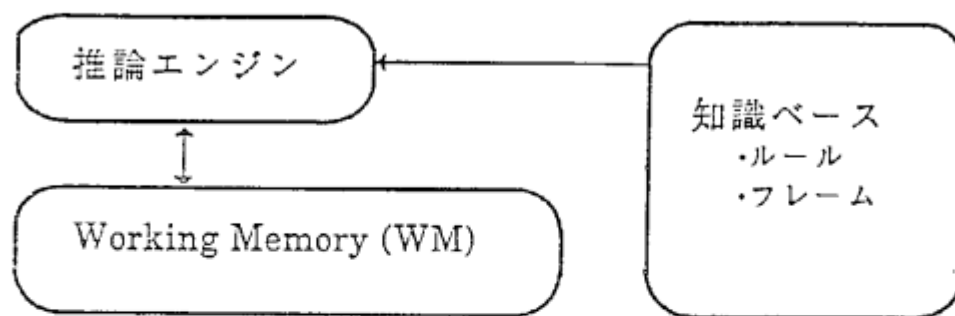


図1.従来のエキスパートシステムによる問題解決

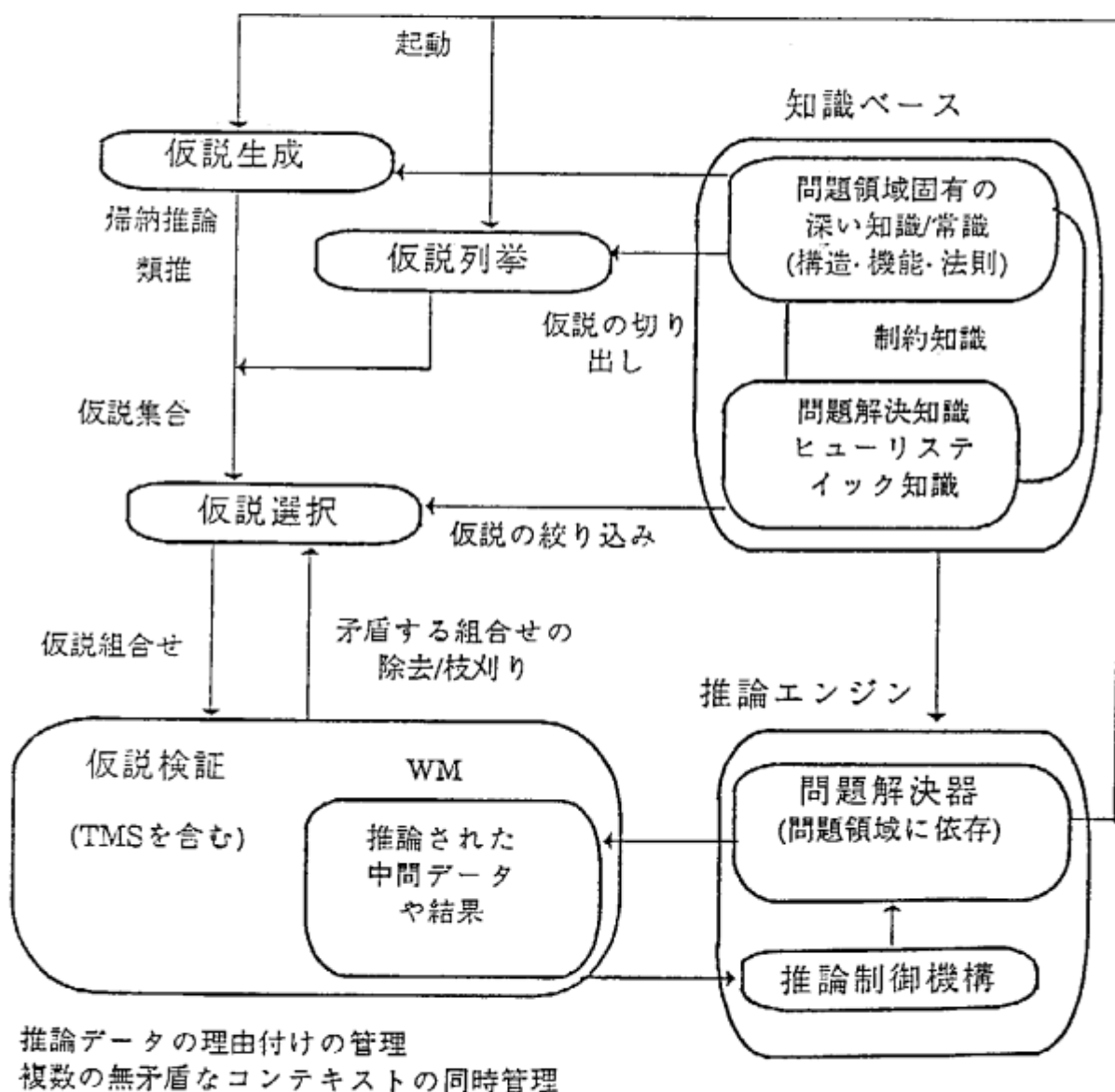


図2. 仮説推論を利用する問題解決 (APRICOT)

6. ATMSの設計型問題への適用

飛鳥井 正道
(キャノン 情報システム研究所)

仮説推論システムのイメージを、de KleerのATMSに基づいたTruth Maintenanceを中心とした観点から述べる。まず、仮説推論を実際問題に適用するときの問題点をあげ、これらの問題点を解決するためにシステムに期待される機能を述べる。

【実際問題への適用時の問題点】

診断型問題を中心として仮説推論の適用例はいくつかあるが、設計型問題への適用はあまりない。ここでは、設計型問題への仮説推論の適用時の問題点について考察する。

◎設計型問題とは

東工大の小林[1]によると、設計型問題（計画型問題を含めて合成型問題）とは、「システムの特性」が与えられたとき、これを実現するような「システムの構造」と「構成要素の特性」を決定することである。

クラス1:

システムの特性 → システムの構造、構成要素の特性

上記の設計問題を「クラス1」の問題と定義すると、より特殊化された問題として、「クラス2」と「クラス3」の問題がある。

クラス2: (レイアウト、スケジューリング問題)

システムの特性、構成要素の特性 → システムの構造

クラス3: (定型的設計問題)

システムの特性、システムの構造 → 構成要素の特性

クラス2の問題は、構成要素を空間的または時間的に組み合わせることによりシステムの特性を満足させる問題であり、レイアウト問題やスケジューリング問題がこれに相当する。クラス2の問題では、組合せ的爆発の回避をどのようにして行なうかが大きな問題となる。

クラス3の問題は、システムの構造の各部分に対して、競合する構成要素の特性の中から最も適切なものを選択することによりシステムの特性を満足させる問題であり、パラメータ設計などのいわゆる定型的設計がこれに相当する。

◎設計型問題における仮説

クラス2の問題における仮説とは、構成要素の空間的または時間的な結合であり、クラス3の問題における仮説とは、競合する構成要素の特性の中からの選択である。

例えば製品の組立ラインの設計問題は、組立作業の各サブラインへの作業配分と、各サブラインでの部品の組立順序の決定を行なうというクラス2の問題である。ここでは、各サブラインへの作業配分および

各サブラインでの組立順序の決定が仮説となる。また、論理回路の簡約化の問題は、与えられた回路に簡約化のためのルールを適用して、最終的に最小な回路の構成を求めるというクラス2の問題である。ここでは、競合する簡約化ルールの集合からの簡約化ルールの選択、より正確に言えば簡約化ルールとそのルールを適用するゲートの集合を選択することが仮説である。

次に例えばレンズ設計の後期段階では、システムの特性をより良く満足するような最適なガラスの組合せを求めるというクラス3の作業を行なう。この作業では「第～レンズのガラスを～にする」という構成要素の特性の選択が仮説となる。また、レンズ設計の初期段階では、何枚かのレンズの合成系としての群に基づいて、システムの特性をより良く満足するような最適な骨組み構造の構成パラメータの値を求めるというクラス3の作業を行なう。この作業では「第～群の焦点距離を～にする」というような構成パラメータの値の決定が仮説となる。

◎設計型問題における仮説推論の問題点

1) 仮説間の依存関係

設計型問題においては、ATMSなどにおけるような平板かつ均質な仮説の表現では限界がある。

例えば前述した論理回路の簡約化の問題では、競合する排他的な簡約化ルールの存在により、簡約化ルールの適用順序に簡約結果が依存する。また、組立ラインの設計問題では、組立部品間の組み付け順序の半順序関係の制約により、仮説に順序情報が必要となる。

また、設計問題では対象が階層的な構造を持つために、階層設計におけるように様々な階層レベルで仮説をたてることにより設計を行なうことが多い。前述したレンズ設計のガラスの組合せ探索や骨組みパラメータの値の決定などを単独で行なう場合には、同一レベルの仮説のみ扱うことが多いために仮説間の階層レベルの問題はあまり生じない。ところが、特に設計の中期段階ではレンズ系の様々な階層レベルの仮説に基づいて設計が行なわれ、これらの階層レベルの異なる仮説間に組合せ的制約や数値的制約などが存在することが多い。例えばある骨組みパラメータの値を仮定した場合、この値を満足するようなレンズ構成の仮説は何種類かに限定されてしまう。また、組立ラインの設計問題の各サブラインへの作業配分の仮説と、各サブラインでの組立順序の仮説も階層レベルの異なる仮説の例である。

このように設計型問題では、仮説間の順序関係や階層関係などの様々な制約関係が存在し、これらをどのように表現するかが問題となる。

2) データの非単調性

ATMSなどにおいては仮説は基本的にはそれぞれ独立であり、仮説の組合せにより矛盾などが生じない限りデータは単調に増加する。従って、ある環境(environment)で成立するデータは、その環境のsupersetの環境で必ず成立するため、データがどの環境で信じられているかが極小な環境の集合であるラベルとして効率良く管理されている。

ところが例えばレンズ設計では、構成パラメータなどを変更した後、CADでシミュレーションなどを行なうと、その構成パラメータに関連する大量のデータが変更され、旧データは消滅する。このようにsupersetとsubsetの環境でデータの非単調性が大きい問題をATMSのような枠組みで記述することは適切ではない。

3) 外部データの取扱い

データが従来システムに存在する場合、理由付け(justification)などの仮説推論のための情報をどのように管理するかが問題となる。通常はCADの修正は困難であるため、CADの外部に情報を持つことになる。例えば仮説推論の例ではないが、通常レンズ設計ではCADに設計データが存在するために、我々の

レンズ設計のエキスパートシステムの旧システムでは、フレームを実現するとき、CADデータとフレームデータを別々の計算機で管理していたことがある。この時には、計算機間の通信やデータの整合性の問題などが発生した経験がある。

【仮説推論システムに期待される機能】

ここでは、上記の問題点を踏まえて仮説推論システムに期待される機能を、Truth Maintenanceを中心とした観点から述べる。

◎仮説と矛盾

仮説推論においては、仮説の組合せによる探索空間の爆発が問題となる。このため仮説としては、ATMSのように明示的に宣言されたものを中心として、仮説の数を必要以上に大きくしないことが重要である。

また、同様な観点から矛盾の数はできるだけ大きく、ATMS的に言えばよりsubsetであることが望ましい。従って、矛盾はシステムが論理的に導出可能なものだけでなく、制約条件の違反など問題解決機構が明示的に与えるものを含めるべきである。ただし、矛盾として制約条件の違反などを含めると、推論の過程などで制約条件の範囲が動的に変更されることが考えられ、このときの矛盾の取り消しなどをどのように扱うかが問題となる。ATMSのように矛盾を極小な環境で効率良く表現するシステムでは、単に矛盾データベースから取り消したい矛盾を取り除くだけでは不十分であり、取り消したい矛盾環境が包含しているsupersetの矛盾環境が存在する場合には、その環境を復元する必要がある。

◎探索空間と探索

ATMSに基づいた仮説推論では、仮説の数が増えると環境の数が飛躍的に大きくなる。このような広大な探索空間から効率的に解を求めるためには、

- 1) 仮説間の依存関係などによる探索空間の限定
- 2) 探索空間の効率的な探索が重要である。

1)は探索空間を限定するための拘束条件的なものであり、2)は探索空間を効率的に探索するための優先条件的なものである。以下では、探索空間の限定と効率的な探索について考察する。

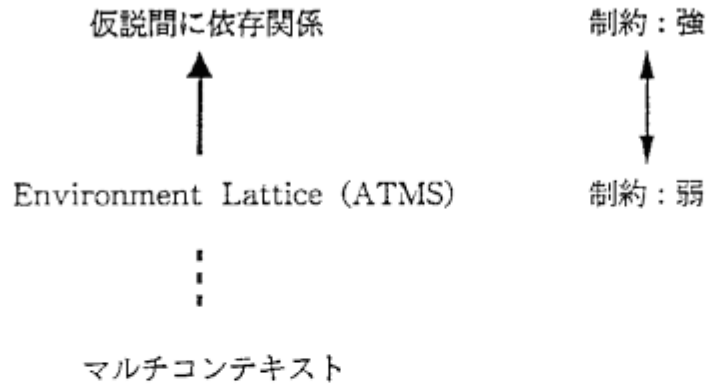
◎探索空間の限定

ATMSでは仮説を宣言すると、それらの仮説の組合せにより探索空間としてEnvironment Latticeを考えることができる。探索空間をできる限り小さくし、探索の負荷を減少するためには、仮説間に依存関係などを記述することにより、探索空間の構造を自由に表現できることが必要である。

〔図1〕に示すように、de KleerのATMS[2]におけるような仮説の連言を中心に考えると、仮説間に様々な依存関係を記述することにより、より限定された探索空間が表現できる。このような依存関係としては、仮説間の矛盾の関係[2]、仮説間の選言や排他的選言や否定の関係[3]がある。また、例えば仮説に優先順位を設けることにより探索制御などに利用することや[4, 5]、仮説にその仮説が置かれている状況に関する情報を付加することにより仮説間に順序関係などを記述できるようにすることや[6]、仮説に階層関係を記述することなどが考えられる。また他方では、仮説を全く導入しない単に複数のコンテキストを持つマルチコンテキストの構造も考えることができる。

◎効率的な探索

ATMSに基づいたシステムにおいては、再計算などの冗長な計算を避けるための極小な環境での推論の実行の優先、および矛盾を早期に発見するために矛盾を導き出す知識の優先的な適用などの推論制御[7]



〔図 1〕 探索空間の限定

が最低限必要となる。また、縦型・横型探索のみならず、仮説の重み付けなどによる様々な探索戦略をどのように実現するかなどの問題もある[4、5、8、9、10]。

◎その他

ATMSは仮説の組合せに基づいて知識を階層化していると捉えることもできる。このように考えた場合、ルールなどの推論の知識を意味的な近さにより階層化する黑板モデルなどにおける手法、対象の知識を意味的な近さにより階層化するフレームなどの知識表現など、他の階層化手法とどのように結合するのか、またわかりやすいユーザ・インタフェースをどのように設計するかなどの問題がある。

【参考文献】

- [1] 小林, 問題解決と類型的タスク, AIセンタ知識情報処理調査委員会知識システム方法論WG資料, (1987)
- [2] de Kleer, J., An Assumption-based TMS, Artificial Intelligence, Vol.28, pp.127-162 (1986)
- [3] de Kleer, J., Extending the ATMS, Artificial Intelligence, Vol.28, pp.163-196 (1986)
- [4] 董, 中川, 不確実な知識におけるATMS, 情報処理学会知識工学と人工知能研究会資料, Vol.52, No.1, pp.1-8 (1987)
- [5] 劉, 西田, 堂下, 不確実な情報によるATMSの制御, 日本ソフトウェア科学会第4回大会論文集, pp.127-130 (1987)
- [6] 董, 中川, assumptionとsituation, 日本ソフトウェア科学会第4回大会論文集, pp.119-122 (1987)
- [7] de Kleer, J., Problem Solving with the ATMS, Artificial Intelligence, Vol.28, pp.197-224 (1986)
- [8] de Kleer, J. and Williams, B.C., Back to backtracking : Controlling the ATMS, Proc. AAAI -86, pp.910-917 (1986)
- [9] 井上, 導出を用いた仮説探索, 情報処理学会第35回全国大会, 1567-1568 (1987)
- [10] 井上, 仮説推論による探索木の枝刈りについて, 日本ソフトウェア科学会第4回大会論文集, pp.131-134 (1987)

7. VLSI 論理設計における仮説推論

丸山 文宏
(富士通研究所 システム研究部)

1. 仮説推論に対する期待とイメージ

(1) 再設計の効率的実行

設計問題に仮説推論を導入することにより、設計及び環境の変化に伴う再設計を効率的に行うことが可能になる。環境とは、第一に、制約条件である。ある制約条件を緩めれば、今までその制約条件のために選択できなかった（しかも、他の評価尺度に関してはベターな）選択肢を選ぶことが可能になるかもしれない。逆に、制約条件を強めれば、設計変更を余儀なくされるかもしれない。その他の環境とは、ひとつの設計対象に注目した時の、まわりの設計状況である。例えば、隣りの大きなブロックのために選択できなかった選択肢でも、そのブロックが小さく設計し直されれば、今度は選ぶことが可能になるかもしれない。

(2) 不完全な情報の取り扱い

仮説を用いて不完全な情報を取り扱うことができる。不完全な情報として考えているのは、次のようなものである。例えば、自動設計で、システム自身ではこれ以上設計法が見つからない場合、システムから見れば「オルタナティブはこれ以外にない」ことになり、他の部分を再設計の対象として選ぶことになるだろう。この場合、「オルタナティブはこれ以外にない」ことが、他の部分を再設計する理由となる。しかし、他の部分を再設計しても結局全体の制約条件を満たすことに失敗することもある。この時、ユーザは他の効果的な設計法を知っているかもしれない。このような場合には、ユーザの設計法を利用することはもちろん、「オルタナティブはこれ以外にない」を取り消したい。なぜなら、他の部分を再設計する理由として、「オルタナティブはこれ以外にない」がはいってしまっているから。

2. 仮説推論のターゲット

2.1 ターゲットのレベル

アプリケーション・レベルをターゲットとし、アプリケーションに依存する要素もまずは一緒に考えていく。

2.2 適用領域

VLSI 論理設計のためのエキスパートシステム

2.3 仮説として何を選ぶか？

可能な design choice (選択枝だが、結果としての設計例で表現する) の各々を仮説とする。例えば、32ビット加算器を設計するchoiceとしての、4ビット・キャリールックアヘッド・アダー8個と2ビット・キャリールックアヘッド・アダー16個。

この他、制約条件が強化あるいは緩和されて繰り返し実行されることを想定し、制約条件も仮説として扱う。また、上に述べたように、「オルタナティブはこれ以外にない」という情報も仮説として扱う。すなわち、ユーザによって新しい設計法が提示された場合には、この仮説をOUTにする。

2.4 仮説推論の結果生じる矛盾とは何か？

設計された回路が与えられた制約条件(回路規模、スピード等)に違反すること。

3. 仮説推論の問題点

3.1 仮説選択における問題点

(1)どのように仮説を選択するか？

(2)環境の変化に対して、前に放棄した仮説をどうやって生き返らすか？

3.2 評価における問題点

(1)制約条件に対して評価できるようになった時点でなるべく早く評価を行う。

3.3 仮説取替(再設計)における問題点

(1)どの仮説を放棄するか？

(2)取り消す必要のないデータをどのように残すか？

4. 仮説推論に対するアプローチ

基本的には、導いた設計データに対して、その導出された理由を付加しておく。例えば、あるコンポーネントを設計する理由付けはその親にある。親のコンポーネントが不要になってしまった場合には、存在理由が消滅する。この理由付けを表現するためには、設計の階層構造が有効である。

制約条件に対する違反が検出された時には、制約条件も含む失敗の理由付け (Nogood Justification) を生成して、それに基づくバックトラックを行う。再設計する箇所（放棄する仮説）を決定するためには、分野固有の知識を利用する。

5. 参考文献

次の文献は以上の考え方を述べたものである。

「協調型論理設計エキスパートシステム」電子情報通信学会人工知能と知識処理研究会

A I 8 7 - 3 2, pp. 1-8, 昭和 6 3 年 2 月

8. 機械設計における仮説推論

今村 聡

(機械技術研究所 システム部)

1. 機械設計における仮説推論に対する期待

非決定的選択を行う場面は多く、解空間の効率的な探索、データの無矛盾性の管理などに期待する。

2. 設計における仮説の生成、選択、検証

機械設計における、仮説の取扱いについて考察してみる。

仮説生成

1) 部品

- カタログからの候補の選択
- 過去の設計例からの選択（そのまま流用できる場合と、パラメトリックな設計変更を含む場合がある）
- 新規設計

2) 構造

- 設計例からの選択
- 部分的組み合わせによる新構造の提案
- 新規設計

3) 数値

- 制約（連立不等式など）を満たす任意のもの

仮説選択

- 評価関数による順位付け（コスト、性能、技術水準、etc）
- 過去の経験による順位付

仮説検証

- 技術計算（性能、寿命、各種の力学計算、etc）
- 解析計算（有限要素法、機構シミュレーション、etc）
- 成立しうるか（形状干渉、各種制約、etc）
- 実験

以上にみるように、機械設計で扱われる仮説はいろいろなレベルのものがある。特に構造を持つものは一階述語論理式では扱いにくく、objectにより表現するのが良いと考える。また仮説自体を詳細化していく過程が設計だともいえる

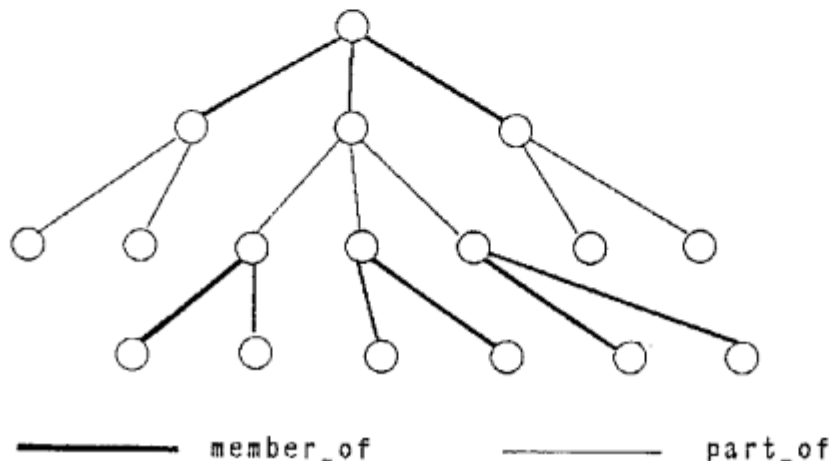
ので、仮説表現の柔軟性が要求される。具体的には、変数を含んだ表現である必要がある。

2. 機械設計における探索の特徴

- 1) 巨大探索空間
- 2) 探索ルール（設計知識）が多様かつ膨大
- 3) 設計データが構造を持つ
- 4) 形状を取り扱う必要がある

3. 解空間探索のイメージ

以上の考察を踏まえて、下図のような機械設計向けの探索方式を提案する。

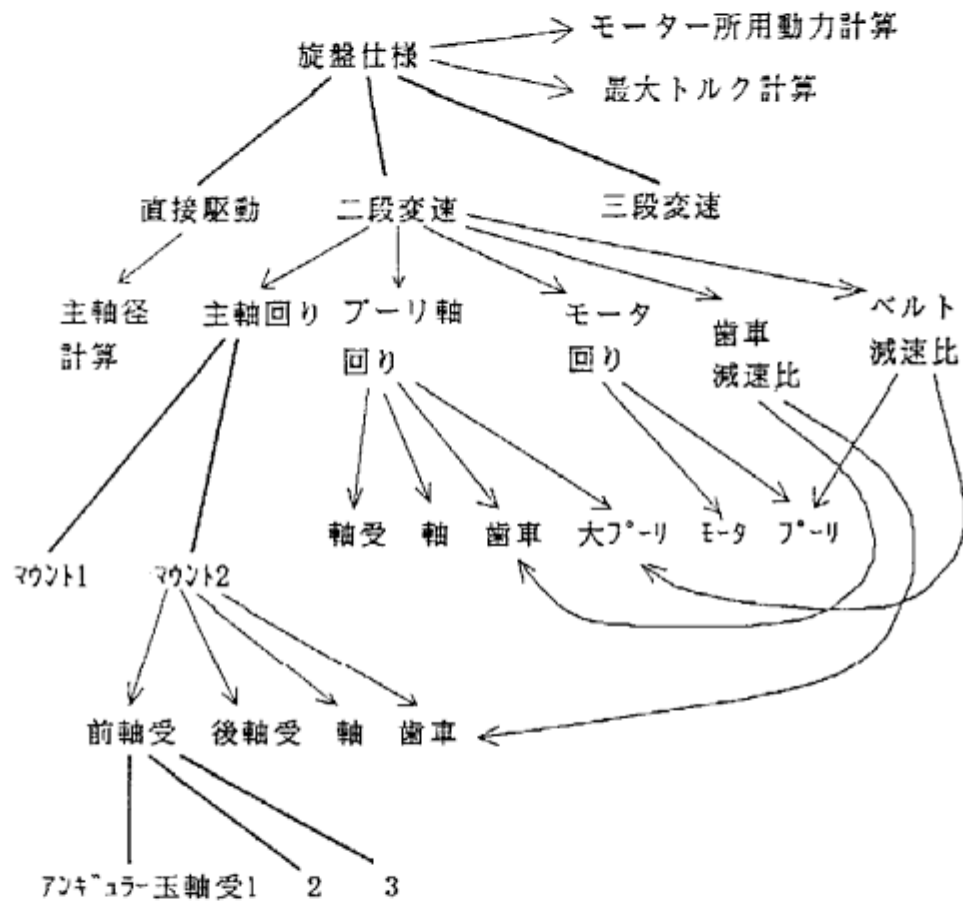


alternativeに選択しうる仮説をmember_ofで表している。仮説が構造を持つ場合は、part_ofにより構成物を表現する。各構成物自体もalternativeな仮説の中から選択しうるとする。このように、member_ofとpart_ofの枝が交互に生々される探索木により、設計解空間を表現する。一つの仮説の構成物間には相互作用が存在し、それを仮説自身が管理する必要がある。この部分（仮説の構成物全体）の解探索には仮説推論の長所を生かせるように思える。

また、構成物の区分の仕方は視点により異なる。普通は、一通りの区分方法では対象を表現しきれず、複数の視点が必要となる。例えば、部品、機能、電気系、機械系、光学系のような視点がある。

次ページに、例として旋盤設計における解探索を示す。ここでは、「二段変速」の構成を部品と機能の二つの視点から行っている。それぞれが共通の構成物を持つ場合がある。また、「ブリー軸回り」、「モータ回り」など、member_of分岐を持たない場合があるが、これは解候補が一つしかない場合である。

旋盤設計における解探索のイメージ



9. 工作機械の主軸設計への仮説推論の適用

加藤 文之

(松下電器産業 東京研究所)

1. 仮説推論に対する期待とイメージ

(1) 仮説推論のイメージ

あいまいであったり、非決定的であったり、互いに競合しあったりしているために矛盾を導く可能性のある知識ベースを利用したとしても、論理的一貫性を維持する事が可能であり、さらにその上で推論を実行する事ができる高次推論の枠組みである。

システムは仮説を生成し、その真偽を判断しながら推論を進める。すなわち観測事実と矛盾しない状態を保つ事により結論を得る。制約条件は仮説の生成や真偽の判断に用いられ、制約条件に反する場合には矛盾が発生したものとして、矛盾の原因に応じた処理が行われる。

(2) 仮説推論に対する期待

実用的なエキスパートシステムを構築するためには、極めて膨大な知識を必要とするであろう。さらにこの知識は、知識獲得の技術が確立されない限り、あいまい性や不完全性を伴うものである可能性が高い。(この問題は知識獲得の技術が確立されたとしても解決困難な性質のものかもしれない)したがって知識ベースの論理的一貫性を維持する事が可能な仮説推論を、知識ベースの整理・検証に有効に用いる事ができるようになる事が期待される。これにより仮説推論は、大規模データベースの管理・保全に関する重要な要素技術の一つとなる。また知識量が増えれば増えるほど、個々には正しくとも組合せると互いに競合するような知識が増えるものと思われる。このような知識ベースを相手に論理的一貫性を保ちつつ結論を得る事が可能な仮説推論は、次世代エキスパートシステムの中心的な要素技術となる事が期待される。

また対象問題の違いによって考えてみると、まず分類型問題(分類・診断・分析等の問題)に関しては、分類結果が複雑な組合せとなるような問題においては、その組合せの候補を仮説として扱う事により、効果的に探索を行う事ができるようになる事が期待される。そして設計型問題(設計・計画・合成等の問題)に関しては、その基本的な推論の枠組みが生成・検証過程に基づくものであるから、仮説推論における仮説の生成・検証過程が、問題解決のための枠組みとして効果的に利用できるようになる事が期待される。

そのほか以下のような仮説推論の利用方法が実現される事が期待される。

まず探索の制御に柔軟性を持たせる事を目的とした利用ができるようになる事が期待

される。例えば制約条件と仮説とをなんらかの形で統合して管理する事により、制約条件が複数存在する場合に緩和すべき制約条件を提案してくれたり、より適切な解を容易に選択する事ができたりするようになれば、問題解決における効率化に非常に有効である。

工程設計等においては、時間順序が非常に重要な役割を担う。したがって仮説推論において時間的關係を有効に扱う事ができるようになる事が期待される。これが可能になると、仮説推論の適用範囲が非常に広がる。

ある仮説が、例えばパラメータのようなものだったとする。そしてその仮説の生成可能な範囲が非常に広い上に、どの程度まで生成範囲を絞ったら良いのかも判断できないというような場合、その仮説の生成範囲そのものを仮説として扱うなどして、仮説生成範囲の暗黙的な制御が可能になる事が期待される。(メタ仮説?)

自動設計も重要ではあるが、ユーザが設定した条件や基本案に対応するために、その条件や基本案に沿って推論を行い、ユーザの意図を実現するためには、どのような策をとったら良いのか、どのような変更を行えば良いのかを提案してくれるようなシステムを実現する事も重要である。このようなシステムに仮説推論を有効に利用できるような事が期待される。

2. 仮説推論のターゲットに関する問題点及びアプローチ方法

(1) ターゲット

現在のところ工作機械の主軸設計問題を対象としている。この問題の設計対象は旋盤の主軸部分であり、ワーク(被切削物)を取付けて回転させる主軸、動力源であるモータ、モータからの回転を伝えるためのベルト、プーリおよびプーリ軸、回転軸を支えるためのベアリング、主軸の回転数を高・低の二段階に変速するギア等からなる。

この問題は、切削対象となるワークの材質、最大ワーク径、主軸の最高回転数、切削時の切込み深さ、ワークの送り速度等の設計仕様を与え、これらの設計仕様を満足するように、各部の寸法、パーツの品番等のパラメータを決定する問題である。

(2) 問題のレベル

工作機械の主軸設計を行うエキスパートシステムというアプリケーション・プログラムを作成するための要素技術として仮説推論の枠組みの適用を検討する。

機械設計においては、制約条件が複雑にからみあっているため、ある設計パラメータについてはその値を仮定して設計を進め、デッドロックに陥らないようにしなければならない場合がしばしばある。このような場合、人間の設計技術者は、過去の設計事例等を参考にして経験的な仮定を行って設計を進めるが、制約条件が互いに関係しあっているような場合には、準最適解を得る事は非常に困難である。

そこでルーチン・デザインにおけるパラメータ・チューニングを仮説生成・検証過程としてとらえ、仮説推論の枠組みによる問題解決を試みる事にした。この問題における仮説は設計パラメータであり、これにはプーリ比やギアの歯数等がある。なお設計パラメータは、JIS規格の利用やカタログからの選択等を行う事により、すべて離散的で有限個数のものとして扱う事にする。またこの問題においては、設計仕様に不適合あるいは作成不可能となる事を矛盾としている。

(3) 課題・問題点

制約条件を厳しくしすぎると解が得られなくなってしまうたり、逆に制約条件をゆるめすぎると、境界条件ぎりぎりの解なのか、それとも余裕がある解なのかが問題になっ

たりする。したがってある程度の制約条件下での解を比較して、その中から適切な解を選び出す事ができた方が望ましい。(ある程度というのが実は極めて難しい。厳しい制約条件から徐々に制約を緩和して行き、適当な個数の解が出たならばその中から選ぶ、という事になるのであろうか。したがって仮説推論とは直接の関係は浅いかもしれないが、制約条件の緩和は極めて重要な課題である。)また設計の具合によっては、設計途中で制約条件を絞り込んだ上で、異なる制約条件下での解を比較する事ができる方が望ましい。

このような事項を考慮すると、複数の設計案を同時に扱う事(multiple context)が可能であり、さらに、異なる制約条件下で得られたそれぞれの設計案の集合の間で、互いに切り換えを行う事(context switching)が可能な枠組みが望ましいという事になる。しかしmultiple contextを扱うと、機械設計においては通常は仮説として扱う設計パラメータの生成可能範囲が非常に広いものとなり、簡単に組合せ的爆発が生じてしまう。

以上より、次のような課題が出てくる。

- ・ multiple contextを扱う事が可能な仮説推論の枠組みの実現
- ・ 上記枠組みにおけるcontext switchingの実現
- ・ 組合せ的爆発の回避

(4)アプローチ

前項で述べた課題から、TMS的な仮説推論の枠組みよりもATMS的なものの方が適しているのは明らかである。ただし対象としているのがルーチン・デザインであるため、設計手順そのものはあらかじめ決まっている。したがってここではATMSの枠組みそのものを利用するのではなく、手順というものを生かして順次仮説を追加生成し、その都度仮説の組合せを展開して行くという枠組みを用いる。すなわち問題解決に必要なすべての仮説の組合せをあらかじめ一度に展開しておくのではなく、必要に応じて仮説を追加して行くという方法を用いる。具体的には、仮説の組(environment)の集合を管理しているpoolと呼ばれるオブジェクトを、generatorと呼ばれる仮説生成機構およびtesterと呼ばれる仮説検証機構へ順次通す事により設計を進めて行くというgenerate and testの手法に基づいた枠組みを用いる。environmentは、generatorによって並列に展開され、そしてtesterによって矛盾を含まないものだけが残される。

このような枠組みによれば、pool内にはその時点において必要なenvironmentのみが存在する事になるため、組合せ的爆発を軽減する事ができる。さらに設計過程においては、制約条件を有効に利用し、generatorにおける仮説生成範囲の絞り込みを行う事によって、組合せ的爆発を軽減する事が可能である。

またこの枠組みは、仮説の検証が行われた後は、矛盾を含まないenvironmentだけが残されて行くので、基本的にはbacktrackingが生じないシステムとなる。しかし処理を行うpoolを変更する事により(context switch)、backtracking的な処理も可能である。もちろんあるpoolが管理している複数のenvironmentは互いに独立であり、それらのうちのどのenvironmentでも同様に、自由にアクセスする事ができる。すなわちこの意味でのcontext switchは極めて容易である。

いくつかの部分問題に分割できる場合には、それぞれの部分問題にpoolを割り当てておいて処理を行い、それらのpoolをmergeする事により、全体の結果を得る事も可能である。これによりgeneratorが一度に取扱わなければならないenvironmentの数を減らす事ができるので、計算機の負荷を軽減する事ができる。また部分問題において解が絞り込まれる場合には、特に組み合せ的爆発を回避するためにも有効である。

以上

10. モデルに基づく診断と 仮説推論の関係

赤埴 淳一
(NTT 情報通信処理 研究所)

1. 仮説推論に対する期待とイメージ

従来の知識処理システムは、演えきをその推論方式の基盤としていたので、不完全な知識を扱うことが困難であった。例えばルールベースシステムで不完全な知識を扱おうとすると、ルールに知識が不完全な場合の処理を記述することになり、ルールの維持管理が困難になってしまう。仮説推論を応用することにより、不完全な知識を扱うことができる推論機構——不完全な知識を仮説で補い、さらに必要な知識を決定し利用者に問い合わせを行って知識を獲得するような推論機構——を実現できないであろうか。

このような推論機構を用いて、次のような機能を持つドメインシェルが実現できるであろう。

【診断型ドメインシェル】

診断対象から得られた情報が診断を行うのに不十分な場合に、診断に必要な情報を決定し、診断対象から情報を得て診断を行う。

【設計型ドメインシェル】

設計案が複数得られた場合あるいは得られなかった場合に、変更すべき仕様を決定し、利用者に問い合わせを行って設計を行う。

2. 仮説推論における問題点とそのアプローチ

《基礎レベル》

問題点: 仮説推論の統一的枠組みの構築

(A)TMS, 発想推論 (Logic-based TMS), デフォルト推論等の関係がすっきりしない。これらを統一的に扱う枠組みが必要。

アプローチ: 述語論理を用いた定式化 (発想推論をベースとする)

例えば、仮説推論を次のように定義する。

Σ : 一階述語論理式の集合 (ファクト・ルールを表す)

Δ : 自由変数を含んだリテラルの集合 (仮説集合を表す)

が与えられたときに、

ϕ : 閉論理式 (説明すべき事実を表す)

に対して

$\Sigma \cup D \models \phi$

$\Sigma \cup D$: consistent

を満たすような極小の

D : Δ の要素の基礎例の集合 (仮説を表す)

を導く推論方式。

この定義のもとで次のようなことが言えないであろうか。

<ATMS との関係>

Problem Solver: Σ を公理系とする一階述語論理の定理証明器

Assumptions: Δ (但し, 自由変数を含まない)

としたとき, D は ϕ の environment(の1つ).

<デフォルト論理との関係>

Reiter の normal default

$$\delta = \frac{\alpha(X) : M\omega(X)}{\omega(X)}$$

は ABnormal 述語を導入して,

$$\alpha(X) \wedge \neg AB(X) \supset \omega(X) \in \Sigma$$

$$\neg AB(X) \in \Delta$$

と表現できる. (より形式的には, $(\{\delta\}, \Sigma)$ の extension と $\Sigma \cup D$ の extension は一致する.)

《ツール事例レベル》

問題点: 仮説推論をどのように応用するか

上記の Σ , Δ を何に対応させるか.

アプローチ: 問題領域に依存

例えば, モデルに基づく診断では Σ を診断対象の構造・動作記述, Δ を故障した構成要素の集合に対応付けている. また, 同じ診断問題を対象としても, Σ を経験的ルール, Δ を観測とするアプローチもありうる.

問題点: 仮説の検証方法

複数の仮説が得られたとき, より適切な仮説を選択するためにどのような事実を付け加えて, 仮説の検証を行えばよいか.

アプローチ: 仮説の選択基準・検証基準を表現できる枠組みの実現

例えば, 診断型ツールでは仮説 (構成要素の集合) の故障確率が最大となるものが選択される. また, 検証に要するコストの総和が最小となるような測定が選ばれる. このような仮説の選択基準・検証基準を表現できる枠組みおよびこれらの基準をもとに仮説を検証する機構が必要.

問題点: 遅い!

従来の枠組みでは仮説選択プロセスと仮説検証プロセスが明確に分かれていた. したがって, 仮説選択プロセスで条件を満たす全ての仮説が選択されるため効率が問題となる.

アプローチ: 仮説選択と仮説検証の融合

仮説選択・検証プロセスは Generate and Test プロセスと考えることができる. 上述の仮説の選択基準に基づいて Generate される仮説を絞り込めることができるのではないか.

1 1. 故障診断への仮説推論の適用

和田 慎一

(日本電気 C & C システム研究所)

1. 仮説推論への期待とイメージ

仮説推論は種々の問題解決で現われる不確かな推論を実現するための技法として期待される。

問題解決では、次のような場合に不確かな推論が行なわれる。

- 得られるデータが不十分なため「大体・・・だろう」という推測をする場合
- 何らかの前提に基づいて推論を行なう場合
- 選択肢から選択すべき各候補について試行的な推論を行なう場合

これらの場合、不確かなことを仮定とし、それに基づいて推論を行なうことになる。推論実行中に誤りや不都合が見出された場合には仮定や仮説を取消して推論を切り替え、別の結論を導くことになる。人間はこのような推論を行なっているが、仮説推論により計算機上で実現されることが期待される。

仮説推論は、システム開発の初期段階で必要とされなくても、それ以後の段階で必要となってくる場合が多いと考えられる。システム開発の初期段階では、問題を簡単化するため、種々の前提をたて、それに基づいて知識の記述を行なう場合が多い。あるいは、例外を考慮せずに知識を記述する場合が多い。人間の専門家が自分でも気がつかずに前提、仮定を含んだ知識を提供することもある。しかし、システム開発の phase が進み、知識ベースを正確で、一般的なものに拡張、修正する場合には、最初に立てた前提が不成立であっても成立する知識を記述する必要がある。このような拡張、修正を実現するためには仮説推論が必要と考えられる。

理想的にはアプリケーションの開発の際に必要が生じたときに、仮説推論を簡単に組み込み、その場に応じた適用形式をとれることが望まれる。

2. 仮説推論の問題点及びアプローチ方法

仮説推論の診断への適用

我々は仮説推論を用いた故障診断システムの開発を進めている。故障診断における次のような推論のために仮説推論を適用する。

症状、テスト結果 $\longrightarrow$ 被疑部分
(故障原因として疑わしい箇所)

この中では「大体的場合、・・・である」というような不確かな推論知識が用いられる。不確かな推論知識は、被疑部分をより狭く絞り込むために有効なものであり、誤りを生じさせない限り利用することが望ましい。このため、通常時は不確かな知識も利用して推論を進め、誤りがあると判明したときにはそれを取消して推論を進める方法が必要となる。このような推論を実現するため仮説推論を導入する。

問題点・課題

(1) デフォルト仮定を優先利用する方式

不確かな知識をより多く利用した推論のためには、その知識の成立を表わす仮定をデフォルト仮定として優先利用する必要がある。このための仮定の制御の方法が必要となる。

また、矛盾が検出された後でも、矛盾解消のために除去するデフォルト仮定を最小限に抑え、矛盾を生じさせない範囲でより多くのデフォルト仮定を利用するように制御する必要がある。

(2) 不確定な推論結果の数値的な評価

仮定を用いて導出された各推論結果の確実度を数値的に評価させたい。その際、矛盾の検出に応じてその数値を更新させる必要があるが、どのように更新させるかが問題となる。

例 $a : 0.6$ $b : 0.5$ $c : 0.8$

$x \leftarrow a \text{ and } b$

の場合、

$x : 0.3 (= 0.6 * 0.5)$

とする。 a , b , c の間で矛盾が検出された場合、

$\text{nogood} \{ a, b, c \}$

a , b , c , x のそれぞれの値をどのように更新させるかが問題となる。

(3) 仮説推論の適用できる範囲

仮説推論が適用できる範囲に制約がある。論理的な推論に対しては比較的容易に適用できるが次のようなものに対しては適用が困難である。

ルールの中でデータ削除を行なっている場合

シーケンシャルな記述

このような場合に対しても適用できる方式が望まれる。

アプローチ

◇仮説推論の適用形式：

不確かな推論知識を何らかの仮定に基づいて成立するものとして表わす。

「x x xであり、仮定 a s mが成立するならば y y yである」

仮定 a s mは推論知識における不確実な条件、前提条件を表わす。

◇仮説推論のタイプ：

TMSのタイプとして、DoyleのTMSではなく、ATMSに近いものを採用する。
複数のcontextを扱うためである。

◇contextの制御の方式：

- より多くのデフォルト仮定を利用する（inにする）

矛盾が検出されていなければすべてのデフォルト仮定を利用する。

矛盾が検出された際は、矛盾を生じさせない範囲でより多くのデフォルト仮定を利用する。

- 複数のcontextを同時に扱う

矛盾が検出された際、矛盾を導く仮定が一般には複数個存在することから、矛盾を解消するcontextもそれに応じて複数個となる。そのような複数のcontextを同時に扱う。複数のcontextを取り扱う際、contextに依存せずに成立するnodeと依存するnodeが区別できる。

- 仮定の確度を表わす数値の設定

仮定に確度を表わす数値を与えて、仮定の和によりcontextの間で比較を行なう。

- alternativeな仮定の処理

alternativeな仮定の組み合わせに応じてcontextを生成する。

→より効率の良い処理が望まれる。

1 2. 仮説推論のツール・イメージ

進藤 静一
(三菱電機 中央研究所)

仮説推論に対する期待、イメージ、仮説推論をツールに導入する際の問題点を挙げる。

1. 期待とイメージ

イメージ

確定的知識

(その真偽が確定している知識)

仮説

(その真偽が事前に定まっていない知識、その取りうる範囲はわかっているが値そのものは定まっていないパラメータ)

の2種類の知識を元に

制約 (= 矛盾)

(推論で用いられる要素、または、推論で得られる要素の間で成り立つべき関係)

をみたすべく、仮説に適切に真/偽を割り当てる、または、仮説と異なされたパラメータに適切に値を割り当てることによって、与えられたゴールを達成するシステム。

期待

① 知識の記述 (量) の軽減

上のイメージで中心的となる部分は、制約を満たすべく不確定的知識に値を割り当てる処理をどの程度システムが負担するかであろう。全面的にシステムの基本機能として与える場合、TMS方式の如く制約違反が起こった場合に仮説-結論間の依存関係を利用して制約違反を生ぜしめた根本的原因の仮説を覆し他の仮説を信じて推論をやり直す、

または、A T M S方式の如く仮説の真偽のすべての組み合わせをベースに多重世界を構成し制約に違反する世界を削除する、のいずれの方式にせよ、計算量は莫大なものとなる。一方、ユーザーが負担する、即ち、制約違反時の対処方法を事細かに知識として与えておく（例えば、「こういう制約違反をした場合はこのパラメータの値をこう変える」とか）方式では計算時間の縮小するものの知識の記述が大変で仮説推論の仮説推論たる所以が消えてしまう。即ち、どの仮説をどう信じるか、及び、仮説の他の結論への影響に関して、その処理の為のユーザーによる知識の記述量と、計算時間にはトレードオフがある。期待として、上記の点に関して、ユーザーによって与えられる知識と推論履歴または多重世界をもちいたシステムより与えられる処理が融合されることによる、ユーザーの負担の少ない、かつ、計算時間の大きくない処理が望まれる。

② 仮説間の独立性

特にA T M Sの場合、仮説の間に依存関係がないことが前提となっている。そのため、ユーザーは仮説を設定する前に、設定しようとする仮説に間に依存関係がないことを確かめる必要があるが、このことはユーザーの負担になるだろう。知識の表現の中途段階に於いては知識全体の見通しが悪く、独立と思っていた仮説が知識全体を見回せばなんらかの依存関係があったということが起こりそうだからである。期待として、仮説間の依存関係を気にしないで設定できる環境が望まれる（例えば、推論前に、知識全体を見回して仮説間の依存関係の有無をチェックするとか）。

2 ターゲットのレベルと問題点

ツールとしてみた場合について述べる。

① 重たくないツールに

一般に、ツールの知識表現／推論機構に色々なものを盛り込むと、そのうちのごく一部分しか用いてない応用事例を動かしても、ソフトウェア的に色々なオーバーラップが発生し、パフォーマンスが悪くなる。その対処方法として、ツールのある機能が必要な時だけにその処理部が起動されるようにツールの機能の処理部のモジュラリティをなるべく高くし（プラグイン方式）、重たくないツールとしなければならない。仮説推論をツールとして実現する場合、T M SやA T M S方式が取り込まれることになると考えられる。これらの方式では結論間の依存関係や仮説の組み合わせに基づく多重世界を保持する必要があり、その処理はツールの推論エンジンと密接な関係があるので上述のプラグイン方式の実現が困難かもしれない。しかし、依存関係や多重世界の処理は時間／メモリを特に消費するのでツールのパフォーマンスを良くするためにもその実現が望まれる。

② 透明なツールに

ツール（に限らずすべてのプログラミング言語）において要求されることは、そのシンタックスとセマンティクスがユーザーにわかりやすいことである（透明度）。ツールの提供する知識表現形式／推論制御機構が複雑であれば、ユーザーはツールで実現したいことを表現するのに困難を感じるのみならず、システムの振舞い（推論）が理解しにくいのでデバッグに苦勞する。仮説推論をツールに組み込んだ場合、例えば、今迄真であった仮説が偽と判明することによりその仮説を根拠としていた結論が自動的に偽になる振舞い等、ツールが自動的に行う部分が多くなり、ツールの透明度は低くなる。使いやすいツールを実現するためにも透明度の低下を補う機能（結論間の因果関係や今成立している仮説の組み合わせのわかりやすい表示機能）が大切になる。