

導出を用いた仮説探索

井上 久巳

(財) 新世代コンピュータ技術開発機構

1. はじめに

仮説推論は、問題解決の過程で適合する知識や不完全な知識等を取り扱う場合に必要となるものであり、それらの知識を仮説と見立ててそれに基づいた処理を行うことにより進めていく推論の形態をいう。現在、仮説推論として注目されている方式としては、論理的なアプローチに着眼を置く Truth Maintenance System が挙げられるが、これは次の 3 方式に分類できる。
①Justification-based TMS [4]: 与えられた仮説から導かれるコンテキストの一貫性をデータ依存関係に基づいて管理する。
②Assumption-based TMS [1]: 仮説の組合せに着目する点で①の効率を改善している。
③Logic-based TMS [5]: 既知の無矛盾な知識集合のもとで、観測事実を説明できる仮説を、与えられた仮説集合から演繹により選ぶ。これらの研究は相互に関連する部分もあり、CMS [3] では②と③の理論的基礎を与えており、ところでこうした理論的枠組みを問題解決システムの中で利用する場合には「仮説の生成・選択・検証」という形で取り入れるべきであるが、そのような全体像が明確には与えらることは少なかった。そこで本稿では現在研究を進めている問題解決まで考慮にいれた仮説推論システム APRICOT (Assumption-based Problem solver in ICOT) における導出を用いた探索機能と TMS 機能について報告する。

2. 仮説推論 (assumption-based reasoning)

【仮説とは何か?】まず始めに仮説(assumption)に対する定義を次のように与える。この問題に対しては本来十分な議論が必要である。 semantics では特に問題解決の観点から言えば、choice と default hypothesis とでは意味的に異なるが、後で再び触れる。 syntax は次の通り。いま α という知識が真である事が保証されないとき、「 $\neg\alpha$ 」という節 (default logic [7] における :My/α に対応する) を追加して、 Γ を " α を支持する仮説" とする。 Γ は①明示的に仮説として導入され、② Γ 自身によって tautological に支持されている。

【仮説の利用】仮説は、abductive reasoning の立場から言えば、ある観測事象のに対する論理的整合性の取れる説明のために導入される。これは α を支持する節があったときにその支持集合が全て仮説であるようなものを導出によって求めていることと論理的に等価である。このとき、支持集合は ①右辺を α とする implication の左辺を形成し、仮説以外のリテラルを含まず、②論理

的に無矛盾である。この解釈を適用すれば、Assumption-based TMS (ATMS) ではこの支持集合が「最小の」「環境」集合 (label) という形で表現されており、Logic-based TMS では $\neg\alpha$ から goal-driven によって求まる "assumable" な命題の組合せに他ならない。また、 α が仮説よりも先に陽に得られない場合もあるが、そのときもこの支持集合の概念を用いることができる。

【仮説推論システム】以上の仮説の定義と支持集合との概念による仮説の利用を基にしている仮説推論システム APRICOT の構成を図 1 に示す。

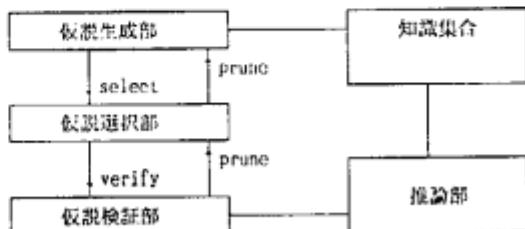


図 1. 仮説推論システム APRICOT の構成

- ・**仮説生成部**：知識の集合から仮説空間を決定し、その中から問題解決において意味のある要素から構成される仮説集合を生成する。
- ・**仮説選択部**：仮説生成部から与えられる仮説集合から、仮説となる要素を候補として選択する。また仮説検証部で得られた結果がフィードバックされて、不要な仮説を選択することを回避する。
- ・**仮説検証部**：仮説選択部で選択された仮説について多項コンテキストに展開し、その正当性を判別するデータ管理機能 (TMS) を有する。

3. 仮説探索アルゴリズム

一般に問題解決の過程は TMS 上で AND/OR グラフとして表現することができるが、この OR ノードを仮説と考えたときのグラフ探索が仮説推論の戦略に相当する。ところが図 1 において、推論部は後述するように問題領域に依存しているので、問題解決の戦略を持つが、TMS は戦略を持たない。そのため TMS 上でのコンテキスト間での焦点は推論部が決定する必要がある。その際、APRICOT の推論部が汎用的な制御戦略として取り入れることができる探索法の概略を示す。

- ①【導出を用いる無評価 AND/OR 型仮説空間探索】 AND/OR グラフは問題解決の過程で段階的に構成されて行く。その際 OPEN リストの先頭から一つずつ solution tree が選択されて展開される。1 つの solution tree [6] は 1 つの無矛盾なコンテキストに相当するが、直接

Resolution Search Strategies for Assumption-based Problem Solving

Katsushi INOUE

ICOT Research Center

点のリストで表現することができる。探索順序はdepth-firstに行うため ATMS とは異なるが、仮説の生成が段階的であり、1コンテキスト当たりのメモリ消費が大きいときは有効な方法である。尚、導出は別のコンテキストの較り込みを行うために用いられる。矛盾の生じた仮説の組合せ(ATMS の nogoodに相当する)はCLOSEDリストに入れられるが、主として導出はこのCLOSEDリストとノード間のAND/OR関係に基づいて行われ、tree上で枝刈りは1つの導出に相当し、葉ノードの枝刈りは ATMS における負起導出[8]に相当する。無矛盾性のチェックは後述するように仮説検証部における Truth Maintenance 機能を本探索法において特にコンテキストの矛盾の発見を効率良く行えるように限定して用いている。

②【評価関数を用いるAND/OR木探索】

仮説間に評価が与えられるときは論理の完全性は保証されないが、best-first searchによって、組合せ的爆発を抑えることが期待できる。この場合の探索法には從来から研究されているゲーム木探索や組合せ問題に適用されているB&B等を取り入れることが可能である。

4. Truth Maintenance 機能

Truth Maintenance 機能はコンテキストを離れて意識すれば、3. ①の探索機能でも行っている。これはコンテキストが推論部にとって意味があるためである。これに対して仮説検証部でのデータ管理は推論部がその時点できめ細やかにコンテキストにおける特定のデータの支持集合を見付けることを主要タスクとする。但し一つのデータは複数のコンテキストに含まれてもよいため、支持集合を求めるときは他のコンテキストにも波及する。よって APRICOT における TMS 機能は ATMS(CMS) 的である。

【支持集合戦略を用いる導出による仮説集合の発見】
支持集合戦略[8]は導出の適用を制限するためのものであり、「公理」間の導出は行わず導出形は「支持集合(set-of-support)」に含める。ここで混亂を避けるためにデータの仮説による支持集合(2. 参照)を支持仮説集合と呼ぶことにし、戦略における set-of-support と区別する。いまリテラル(節でもよい) C の支持仮説集合を求めたいとする。Σ を和項からなる satisfiable な節の集合、Π (C Σ) を C を含む全ての節集合とする。以下の手続きは Π を支持集合とする支持集合戦略を基本的に用いている。本手続きは段階的に Σ を構成する場合に効率が良い。《Step-I》 Π をすべての素包含項(主項)[3]を含んだ節集合に書き替える。《Step-II》 Π と Σ-Π から節の導出を行う。その際、① C が導出形に残るか、② ~C を含む節の他のリテラルの集合が C を含む節のリテラルの集合を包含する、のいずれかである場合に導出を限定する。ここで導出がそれ以上行えないときは、Π の各節で C を imply する項をすべて合せたものが C の「最小の」支持仮説集合であり、Π と Σ-Π を含めたものを Σ として手続き終了。《Step-III》 導出形を Π に (C を含まない項のときは Σ-Π に) 含め、それにより包摶される節をすべて Σ-Π と Π から消去する。

《Step-II》へ。

5. 問題領域に依存した推論部

APRICOT では2種類の仮説探索を行う。3. では矛盾を含むコンテキストを枝刈りするアルゴリズムを、また4. ではあるデータの支持集合を求めるアルゴリズムを与えた。前者は後者を用いて実現される。すなわち、負の和項 A → B が ATMS で管理されているとき、前者の探索ではこれを仮説 A と B が同時に存在することが矛盾する(ATMS における nogood[A, B])と見てその solution tree の枝刈りを意味し、後者の支持集合の概念を用いると仮説 A のコンテキストの基で仮説 B あるいは~B の支持を問い合わせた場合、A → ~B により仮説 B が存在し得ないことを意味する。これらの仮説探索は次のような対象とする問題領域に応じた推論戦略と合わせて適宜使って行く必要がある。

【診断向きの Problem Solver】 仮説(hypothesis)はあるコンポーネントが正常であるとみる。観測○が与えられたとき仮説の否定リテラルのみからなる和項を発見できれば、それらの和積正規形を積和正規形に変換することにより故障コンポーネントの組み合わせを発見できる。

【制約満足化 (CSP) 向きの Problem Solver】 仮説(choice)はある変数への値の割り当てとみる。観測○が発見された時点でその観測あるいは否定を支持する仮説を求め、現在のコンテキストとの無矛盾性を調べる[2]。

6. おわりに

APRICOT は仮説推論を問題解決に使って行く場合における基本的な枠組みを提供しているが、推論部は対象とする問題領域に依存しなければならないという考えに基づいている。さらに本稿では触れなかったが、仮説生成についても対象問題から考えていかなければならない部分が多く、例えば領域固有の知識や深い知識を利用する必要がある。また実用化への課題として、①仮説推論が対象とする応用問題の中でどの部分に対して最も有効であるか? ②応用問題の中で仮説や矛盾は何か? ③仮説間の優先順位や順序関係、さらにレベル付けはどのように表現し得るか? ④論理的な完全性と計算量の効率とのトレード・オフをどのように設定するか? 等の点についても十分考慮する必要がある。

【参考文献】

- [1] de Kleer, J., Extending the ATMS, Artificial Intelligence 28 (1986) 163-196.
- [2] de Kleer, J., personal communication.
- [3] de Kleer, J. and Reiter, R., Foundation of ATMSs, to appear in AAAI, 1987.
- [4] Doyle, J., A Truth Maintenance System, Artificial Intelligence 12 (1979) 231-272.
- [5] Finger, J.J. and Genesereth, M., RESIDUE: A Deductive Approach to Design Synthesis, Stanford NRP, Memo NRP-85-1, 1985.
- [6] Pearl, J., Heuristics, Addison-Wesley (1984).
- [7] Reiter, R., A Logic for Default Reasoning, Artificial Intelligence 13 (1980) 81-132.
- [8] Wos, W. et al., Automated Reasoning, Prentice-Hall (1984).