「ＥＳＰによるＡＴＭＳと問題解決器を融合した仮説推論機構」

－ ＡＳＴＲＯＮ －

藤原 達　飯島 勝美（ＮＴＴデータ通信(株)）　井上克巳（ＩＣＯＴ）

１．はじめに

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る。真偽が不明な知識はとりあえず真(仮説)として推論を進め、後に、矛盾する状況が起こった時には、その矛盾の基になる仮説を修正するという、一種の非単調推論を実現する枠組みである。

この推論の結果得られた無矛盾な仮説の集合を解とすることにより、従来の診断や設計といった問題の解決に対し、プロダクション・システムとは異なるアプローチが考えられている。

仮説を基に推論を行うためには、作業記憶内を動的に管理するなんらかの真理維持機構が必要となる。この真理維持機構としては、ＴＭＳ[Doyle 79]やＡＴＭＳ[de Kleer 86a]などが提唱されている。本稿では、そのうち仮説の組合せに基づいて、一貫性の維持を行っているＡＴＭＳに着目し、これとルール・ベースの問題解決器をＥＳＰで実現した仮説推論システム「ＡＳＴＲＯＮ」について報告する。

２．ＡＴＭＳと問題解決器

対象毎に最適な問題解決器を組み込めるようにするためには、知識管理機構（ＡＴＭＳ）と問題解決機構は明確に独立されるべきである。ＡＴＭＳは、多重コンテキストに基づき、知識の一貫性を管理する汎用の機構であり、それ自身は問題解決を行うものではない。そのＡＴＭＳに、実際に問題領域に依存する問題解決機構を、いかに組み合わせて非単調な推論を行わせるかが、仮説推論を応用していくにあたっての課題となる。

ＡＳＴＲＯＮではルールベースの問題解決器を考慮しており、図１に示すように、問題解決器が事実、仮説及びそれらに対する理由付けをＡＴＭＳに与えると、ＡＴＭＳが効率的に全てのコンテキストを決定する。

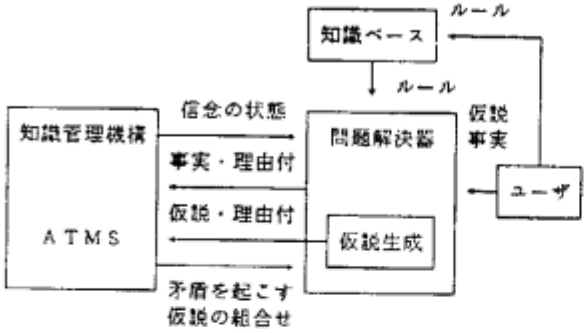

図１．　仮説推論システムの構成

問題解決器はあるコンテキストが矛盾を起こすかどうか、またあるデータがどのコンテキストに含まれるかをＡＴＭＳに確認しながら、推論を進めている。

３．仮説推論アルゴリズム

ＡＳＴＲＯＮの動作アルゴリズムをＡＴＭＳ・問題解決器にわけて説明する。

３．１　ＡＴＭＳ

仮説推論で扱う仮説、矛盾、理由付けを以下の環境、コンテキスト、ラベル等で管理している。

３．１．１　構成要素とその役割

(1)ノード　事実・仮説・導出データ等を表現するデータ構造。
(2)理由づけ　ノードの支持関係を示す。
(3)環境　仮説からなる集合。
(4)コンテキスト　無矛盾な環境の仮説とそれら仮説から推論できるノードの集合。
(5)ラベル　環境からなる集合で、データが究極的に依存する仮説を示す。

ＡＴＭＳは、ノードと理由付けからなる、ネットワークで知識を管理し、
(a)データに対応するノードの生成
(b)ノードの理由付けの付加
(c)ノードのラベル計算
を行っている。

３．１．２　ＥＳＰを用いたＡＳＴＲＯＮの特徴

ＡＴＭＳの多重世界においては、新たな事実の発見等に伴い、理由付けの追加、矛盾の生成といったイベントが次々に発生する。この動きをより適格に表現するため、ＡＳＴＲＯＮではＥＳＰの特徴であるオブジェクト指向をとり入れ、仮説、理由付け、矛盾といった各要素をオブジェクトとして表している。各構成要素のアトリビュート情報を、オブジェクトのスロットに持つとともに、知識一貫性維持のアルゴリズムを、オブジェクト相互のメッセージ交換により実現している。

３．２　問題解決器

本来、問題解決器は問題領域に依存したものであるが、ＡＳＴＲＯＮは特定の対象問題を意識せず、仮説を扱える汎用的なルール型問題解決器を用意している。

３．２．１　構成要素

(1)事実・仮説　ユーザーから与えられたり、ルールの実行に従い、動的に生成される。ルールのマッチングの対象となる。
(2)ルール　条件部は，全てのコンテキストにおいて、それまでに得られた事実や仮説とマッチングを行い、実行部では事実・仮説の追加及びそれに対する理由付けをATMSに与える。
(3)ルール検索キー　データ(事実・仮説)とルールに共通のキーを持たせることにより，マッチングの効率化を図る。

３．２．２　推論アルゴリズム

仮説や事実と、ルールの条件部のマッチングを行う際、ＥＳＰのユニフィケーション機能を利用している。

Assumption-based Reasoning System in ESP - ASTRON -
Toshi Fujiwara, Katsumi Iijima　NTT Data Communications Systems Corp.,
Katsumi Inoue　ICOT Research Center

〈1〉 ルールを，その条件部第一条件の先頭要素をキー値とするハッシュの形で，プロダクション・メモリ(PM)に展開する。
〈2〉 新たに，事実・仮説が、あるアクションの実行やユーザの入力により、言明されるとその先頭要素をキー値として持つルールをPMから取り出す。
〈3〉 〈2〉で抽出した各ルールに対し、言明された事実・ 仮説とのユニィフィケーションをとり、他に条件が無ければそのルールのアクション部をキューイングする。
他に条件があれば、それら残りの条件部に対してユニフィケーションを保存したものを、新たな条件部とするルールを生成し、PMに〈1〉と同様に格納する。
〈4〉 キューメモリを順に実行して実行結果そのものをATMSに送ると，その発火したルールの全ての条件部，実行部を用いて〈条件部〉→〈実行部〉の理由付けをATMSに与える。

4．特徴的な機能

4．1 論理における推論ルール

プロダクション・ルールは Modus Ponens（三段論法）により推論の実行が行われるもので、ヒューリスティックなルールの記述に従来用いられてきた。この枠組みだけで、もう少し論理的な構造のもの（例：回路の構造）を扱おうとすると、論理の完全性が保証されない。

この１つの解決手段として、ＡＳＴＲＯＮではルールに論理的推論ルールの記述を許し、And Elimination, Or Elimination,Modus Tollens等を扱うことができるようにしている。

4.2 仮説の動的生成とDefault推論[Reiter 80]

人間は頭の中で問題解決を行う際、状況に応じて、様々な仮定をしながら推論を行っている。この過程は予め考えられる仮説を全部列挙してから推論するのではなく、必要に応じて仮説を生成し消去する過程と捉えることができる。こういった過程を実現するためにＡＳＴＲＯＮでは、事実から事実を導くだけでなく、事実から仮説を動的に作り出す機能を提供している。これにより、" 必要に応じて" という推論が実現される。

この仮説の動的生成という機能をさらに拡張し、Reiterのnormal defaultルール

$$a(x) : M\ B(x) / B(x)$$

の記述も可能である。内部的には、Ｂを仮説(assumption)とし、ｂを仮説を理由付けに持つ仮定ノード(assumed node)として表現する。Ｂという仮説の存在が矛盾を起こさないなら、ｂという仮定ノードを生成し、$a(x) \wedge B(x) \rightarrow b(x)$ という理由付けとともに、ＡＴＭＳへ手渡す。これにより、矛盾を生じない限り、Ｂをデフォルトの仮説として推論をしていくことができる。

5．実行例 [Default推論の例]

ＡＳＴＲＯＮを用いて仮説推論を「Tweety - Penguin」の例で以下に示す。

＜ルール＞

```
rule1:  bird(X)   → assume (fly(X)).
                          %通常、鳥は飛ぶ

rule2:  pengin(X) → not (fly(X)).
                          %ペンギンは飛べない

rule3:  not(X),X  → contradiction.
                          %論理矛盾
```

＜実行手順＞

```
(1) assert (bird(tweety)).     %Tweetyは鳥であることの
                                宣言
(2) assert (penguin(tweety)). %Tweetyはペンギンである
                                ことの宣言
```

＜推論の推移＞

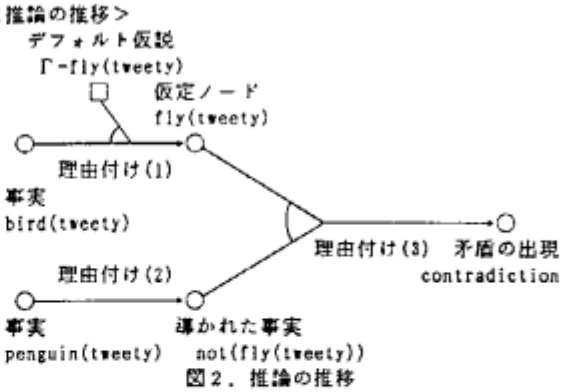

図２．推論の推移

〈1〉 (1)で「TweetyはBirdである」という事実が入力されるとrule1が発火し、「-fly(tweety)∧bird(tweety) → fly(tweety)、という理由付け(1)と共に、「Tweety は飛べる」という仮説を動的に生成し，ＡＴＭＳへ伝える。
〈2〉 新たに、(2)で「TweetyはPenguinである」 という事実が入力されるとrule2 が発火し、penguin(tweety) -> not(fly(tweety))、という理由付け(2)と共に「Tweety は飛べない」という事実をＡＴＭＳに伝える。
〈3〉 〈1〉、〈2〉 の結果からrule3が発火し、理由付け(3) fly(tweety)∧ not(fly(tweety))->contradictionと共に、矛盾ノードを生成する。
〈4〉 ＡＴＭＳは矛盾の解消のため、ラベルをチェックし、「Tweetyは飛べる」という仮説をnogoodにする。

6．終わりに

ＡＳＴＲＯＮは、仮説推論を問題解決に適用する基本的な枠組みを提供したものである。今後の課題としては、

(1) 探索の効率化 [Inoue 88]
３，２，２の推論アルゴリズムで述べたルールの実行部のキューを、スケジューリングする事により、無駄な問題解決手続きの実行を省く。
(2) 制約式の扱い [de Kleer 86b]
積極的に対象問題の仕様や要求条件を扱えるとともに、より効率的な解の探索ができるよう、constraint（制約式）の扱えるConsumerアーキテクチャを導入する。

などがあげられる。また応用として、ＩＣＯＴ第５研究室で取り組んでいる設計問題や計画問題への適用も考えている。

7．謝辞

この研究の機会を与えて下さったICOT 第５研究室 藤井室長,NTT データ通信(株)開発本部 岩下部長，並びに，適切な助言を下さったICOT第５研究室，滝研究員，楢研究員に感謝します。

8．参考文献

[de Kleer 86a] de Kleer,J., "An Assumption-based TMS", Artificial Intelligence 28 (1986), pp.127-162
[de Kleer 86b] de Kleer,J., "Problem Solving with the ATMS",Artificial Intelligence 28 (1986), pp.197-224
[Doyle 79] Doyle,J.,"A Truth Maintenance System", Artificial Intelligence 12 (1979), pp.231-272
[Inoue 88] Inoue ,K.,"Pruning Search Trees in Assumption-based Reasoning",Proceeding of AVIGNON'88,(1988) PP.133-151
[Reiter 80] Reiter,R.,"A Logic for Default-Reasoning",Artificial Intelligence 13 (1980),pp.81-132