

言語処理システムにおける ATMS の利用法について

6C-7

西岡真吾, 堀雅洋, 池田満, 溝口理一郎, 角所収
大阪大学産業科学研究所

1 はじめに

我々は音声理解システムの言語処理部として ASP[1],[2]を開発した。ASP は問題解決において重複した推論をおこなうことがある。ATMS[3]を採用することにより、重複した推論を避け、推論の一貫性を管理し、ASP の効率を向上させることができる。本稿では、ASP のような多様な知識と膨大な探索空間を対象とした問題を扱うシステムに ATMS を導入する際に、有効となる機能の拡張と、その運用法を提案する。2節で、この問題を一般的に考察し、3節において ASP の具体例を用いて説明する。

2 ATMS の改良

図 1 に一般に仮説型推論を用いたシステムの全体図を示す。Problem Solver(ASP) はワーキングメモリのデータを用いて推論をおこない、その結果を ATMS に通知する。ATMS は Problem Solver のおこなった推論の一貫性を管理し、ワーキングメモリには常に無矛盾なデータしかないようにする。Scheduler は Problem Solver が遷移する環境を制御し、遷移した先の環境を ATMS に通知する。

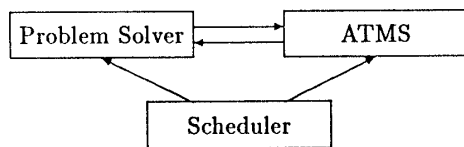


図 1 システムの構成

ATMS は次に探索すべき空間の管理および、これまで発見された矛盾の管理をおこなうが、どちらにも同じ仮定を用いる。

しかし、一般には矛盾の妥当な記述と探索制御の妥当な記述が一致するとは限らない。実際に、ASP においては探索の管理としては候補単語の選択を仮定とすれば十分であるのに対し、矛盾が起こった場合にはその矛盾を候補単語単位で持っている有効な枝刈りがおこなえない。ASP の矛盾は候補単語が持っていた意味情報を用いて定義した方が枝刈りの効率化が図れる。

問題は ATMS が探索すべき空間の管理と矛盾の管理の双方に対して、同じ仮定を用いることにある。以下では、ATMS の枠組みを変更することによって、この問題の解決を与える。

2.1 制御空間と探索空間の分離

以後、意味情報を用いた制約の下で問題解決を管理している空間を探索空間、環境の遷移を管理するための空間を制御空間と呼ぶ。

制御空間の一つの仮定は探索空間の複数の仮定に対応する。したがって、制御空間に属する仮定を用いて枝刈りをおこなった場合、探索空間における矛盾を必要十分に表すことができず、有効な枝刈りがおこなえなくなる。探索空間は矛盾を表すという点などにおいて、制御空間より細かな記述が可能である。逆に、探索空間の仮定を用いて環境遷移の制御をおこなった場合、問題空間の表現が複雑になり、環境遷移の制御が煩雑になる。したがって、環境遷移の管理のために探索空間を用いることは望ましくない。

この問題は、制御空間と、探索空間を区別し、環境遷移の制御は制御空間で、推論(矛盾の管理等)は探索空間においておこなうことによって解決される。つまり、発見された矛盾による枝刈りは探索空間においておこなわれ、ワーキングメモリの内容が無矛盾に保たれる。制御空間は環境遷移の管理のために使われるが、環境遷移が終わった後に ATMS に遷移した先の探索空間を通知しなければならない。このために、制御空間から探索空間への写像が必要になる。この写像は、対象とする問題に依存して用意されるべき性質のものであり、ASP においては、直観的に候補単語から意味情報への写像になる。

このように、2種類の仮定を用いることによって、適切なコントロールと適切な枝刈りを同時に達成することができる。

2.2 仮定間の依存関係

ATMS では問題解決器で用いられる全てのデータ間の依存関係は支持理由と呼ばれる一種の有向リンクで表現される。この支持理由を用いると、原理的には全ての依存関係を表現できるが、仮定間にあらかじめ与えられる依存関係を表現するうえでは、管理上の負荷が大きい。あらかじめ与えられる仮定の包含関係を取り扱うことによって、この負荷を軽減し、必要に応じて矛盾の記録に用いる仮定の柔軟な選択を可能にすることができる。この包含関係の取り扱いは ATMS の通常のラベル計算と同じように、ビット演算によっておこなうことができるので、ATMS の効率はこれによってひどく悪くなることはない。

3 ASP の枝刈り

本節では、2.1, 2.2節で述べた方法を用いることによって、ASP の探索の有効な枝刈りができる例を説明する。

図2のような候補の組が、ASP の入力候補として与えられたとする。左の列から順に第1, 2, 3, 4文節の候補であり、各列から1つの候補を選択することにより全文を確定する。このときASPの動作は次のようになる。

長い	母さんは	手を	洗う
	丸太を	きれいに	なでる
			削る

図2 ASP への入力例*

*これは説明のために作成したものであり、実際の例ではない。

ASP は、まず「洗う」を、次に「手を」を、次に「母さん」を選択する。この時点での部分解は「母さんは、手を、洗う」である。しかし、第1文節の「長い」は「母さん」を修飾できないので、全ての文節候補を選択することができない。そこでASPはこの部分解からは正解が確定できないと判断し、「手を、洗う」という部分解まで戻ってくる。しかし、この部分解からはもはや「長い」、「丸太を」は選択できず、「手を、洗う」という部分解においてもASPは「洗う」という部分解まで後戻りする。ASPは、「洗う」という動詞は目的語として「体の部分」しか取ることができないとしている。そうするとこの部分解からもどの候補も選択することができない。ここでASPは「洗う」にかえて「なでる」を選択する。しかし、ASPは「なでる」の意味情報を「洗う」の意味情報の部分集合としているので、「洗う」を選択したときと同様の過程を経て全候補を選択するのに失敗する。最後にASPは「削る」を選択、次いで順に「丸太を」、「きれいに」、「長い」を選択し、全候補が選択されたので停止する。

もし、「なでる」の意味情報が「洗う」の意味情報の部分集合となっていることが事前に分かっていたら、部分解「なでる」からの探索は部分解「洗う」からの探索の部分となることがわかる。したがって、「洗う」が失敗したことから「なでる」から実際に探索することなく「なでる」が失敗することが予測できる。

2節で述べた仮説型推論を用いてこの探索の管理をおこなう。制御空間の仮定としては候補単語の選択を用いる。実際、ASPのコンテキスト管理は候補単語の選択なので、このように仮定をとることは都合がよい。

ASPでは動詞「洗う」の意味的制約は「動作主(省略可)、目的物[体の部分]を要する」であるので、制御空間の仮定「洗う」に対応する探索空間の仮定を、「動作主(省略可)」、「目的物[体の部分]」とする。同様に「なでる」の意味的制約は「動作主(省略可)、目的物[髪]を要する」であるので、制御空間の仮定「なでる」に対応する探索空間の仮定を、「動作主(省略可)」、「目的物[髪]」とする。これらの関係は前節で述べた写像であり、概念的には図3のように表される。

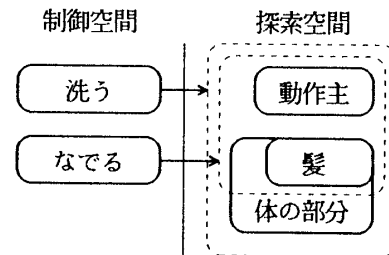


図3 制御空間から探索空間への写像の例

いま、環境{「洗う」}は矛盾である。このとき、既に述べたように環境{「なでる」}は実際に探索がおこなわれることなく矛盾であると判断できる。これは環境{「なでる」}に対応する制御空間での環境{「動作主(省略可)」、「目的物[髪]」}が環境{「洗う」}に対応する制御空間での環境{「動作主(省略可)」、「目的物[体の部分]」}のスーパーセットとなっているとき環境{「なでる」}が矛盾と結論付けられる。これは制御空間の環境から簡単に計算できる。

ここで問題となるのは、「目的物[髪]」と「目的物[体の部分]」の比較である。2.2節で述べたように、仮定の間に関係を持ち込む。ここでは、ASPが言語的知識として持っている「髪」→「体の部分」を用いることによって、先の環境の比較をおこなうことができる。この比較は通常のATMSの仮定のインプリメントと同じく、ビット演算でおこなうことができるので、システムへの負担は小さい。

4 おわりに

本稿では、言語処理システムASPへのATMSの導入について報告した。ここで述べたように、ATMSは形式的に整理された仮説推論の枠組みを提供するが、多様な知識や複雑な探索空間を取り扱う際には、いくつかの問題がある。これを克服する目的でATMSの機能拡張と運用法を検討した。制御空間と探索空間の分離、および両者間の写像の導入によりATMSを用いた新しい問題解決の枠組みが得られた。本稿では触れなかったが、制御空間でのスケジューリングは、ASPの連想機構や制御知識を用いておこなわれている。今後は、探索制御の機構とそのための知識の洗練をおこなう必要がある。

参考文献

- [1] 溝口 他: “音声理解システムの認知科学的検討”, 日本認知科学会第1回大会, B-1 (1985).
- [2] 堀 他: “音声理解システムにおける認知科学的検討—統合化された意味処理のためのフレームワーク—”, 日本認知科学会第2回大会, A-2 (1985).
- [3] J. de Kleer: “An Assumption-based Truth Maintenance System”, *Artificial Intelligence*, 28, pp.127-162 (1986).