

プランニングの効率化の一手法

1C-6

— 関連規則の導出について —

長田 正 ○単蘭娣

九州大学工学部

1. はじめに

プランニングとは与えられた知識に基づいて、初期状態と目標状態を結ぶプロダクション規則の系列を導くことである。一般的に、導かれたプロダクション規則の系列あるいはプランが最適であるかどうかは、プランニングのシステムを評価する一つの基準となる。ここに、最適であるとは、そのプランにムダな規則が含まれていないことを指す。ここでは、システムの評価を高めるため、プランニングに先立って、不完全(変数を含む)であるが、問題解決に関連のある規則(関連規則)を導出する一つの方法を提案し、その具体的なアルゴリズムを示す。

2. 関連規則の導出の出発点としての集合

与えられた問題の対象世界を次のようなモデルで記述する。

$$M = [O, R, P]$$

ここで、Oはobjectの略で対象世界にある対象物(オブジェクト)を表し、Rはrelationの略で各オブジェクトの相互関係を表し、Pはpropertyの略で各オブジェクトの持つ属性を表す。

対象世界のオブジェクトOは、固定されたオブジェクトと可変なオブジェクトに分けることができ、固定されたオブジェクト全体をFixedという集合で表す。対象世界MをFixedを出発点とする構造図で表現すると図1のようになる。図1では、節点はオブジェクトを表し、二つの節点をつなぐ弧線は、

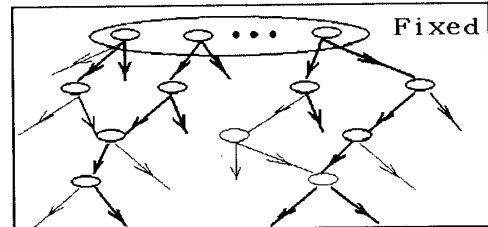


図1 「Maint」の導出

その二つのオブジェクトの相互関係を表す。又、節点より出て他のいかなる節点とも接続されない弧線はその節点に関する属性あるいはかかわる関係を表す。与えられた世界の初期状態と目標状態を表す構造図をそれぞれ F_1 、 F_g とする。 F_1 と F_g の出発点は、共にFixedであることを注意されたい。このような構造図を用いて、初期状態から目標状態への途中で必ず変わる関係Rの要素と属性Pの要素、つまり関連規則の導出の出発点を見出すことができる。この探索過程は次のようである。つまりRについては、 F_1 と F_g をFixedから調べ、同一の関係あるいは属性が見つかれば、それを図1のように太線にする。又、一つの節点に入るすべての弧線が太線になるならばその節点は固定可能であるので、その節点から次の探索を続けていく。そうではないときには、その節点に続く関係が維持不可能になるので、その節点に続く部分を探査する必要がなくなる。Pについては、 F_1 と F_g をすべての節点について調べ、ある節点に関する同一の属性が見つかればそれを太線にする。 F_1 と F_g を調べることにより得

られた構造図の太線は, F_1 から F_G へのプランニングにおいて維持可能である関係あるいは属性だとは考えることができるので, $Maint(maintainable$ の略) という集合で表す。

そうすると, 次の二式が得られる。

$$\neg Maint_1 = F_1 - Maint$$

$$\neg Maint_G = F_G - Maint$$

即ち, 消去されるべき関係と属性の集合である $\neg Maint_1$ と達成されるべき関係と属性の集合である $\neg Maint_G$ の二つの出発点が求まった。

3. 関連規則の導出

ここでは, STRIPS のプロダクション規則が用いられる。つまり規則 R は P (前提条件式), D (消去リスト) と A (追加リスト) からできている。前節で得られた $\neg Maint_1$ と $\neg Maint_G$ をここで, それぞれ $\neg Maint_{10}$, $\neg Maint_G$ とし, $Maint$ を $Maint_0$ とする。例として $\neg Maint_{10}$ から関連規則を導出する方法を考えてみよう。

$\neg Maint_{10}$ はプランニングにおいて消されるべき状態要素の集合であるので, 規則の D に含まれるべきである。したがって, $\neg Maint_{10}$ については, 図2のようにこれを D リストの一部として持つ規則を求めることができる。 P_{10} は R_{10} に対応した前提条件式である次は P_{10} と $Maint_0$ を調べる。

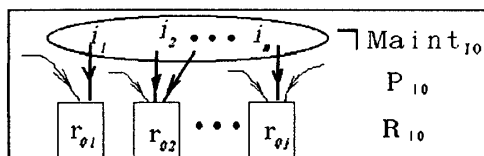


図2 関連規則の導出

もし P_{10} と $Maint_0$ の間に矛盾する状態要素がなければ, $Maint_0$ が維持されるとき R_{10} の規則が適用可能なことを示す。そうではないならば, R_{10} の規則が $Maint_0$ の維持の下では適用不可能なことを示す。しかし R_{10} の規則はプランニングにおいて関連規則であるので, 問題を解決するため, $Maint_0$ には

維持し続けることができない要素が含まれていることが明かになる。ここで, このような要素を $\Delta \neg Maint_1$ で表す。続いて $\Delta \neg Maint_1$ について以上の探索過程を行い, このような計算プロセスを $\Delta \neg Maint_n = \Phi$ まで繰り返し, 結果としての関連規則は $R_{1-Rela} = R_{10} \cup R_{11} \cup \dots \cup R_{1n}$ になる。同じように $\neg Maint_G$ から関連規則 R_{G-Rela} が得られる。

4. アルゴリズムとまとめ

これまでの探索プロセスは次のようなアルゴリズムでまとめられる。

1. Fixed $\rightarrow F$

2. until $\Delta F = \Phi$.

begin:

R について F から F_1 と F_G を調べ固定可能な object $\rightarrow \Delta F$;

維持可能な関係 $\rightarrow Maint_0$

$\Delta F \rightarrow F$;

end.

3. P について F_1 と F_G のすべての節点を調べ, 維持可能な属性 $\rightarrow Maint_0$.

4. $Maint_0 \rightarrow M$; $\neg Maint_0 \rightarrow \neg M$

until $\Delta \neg M = \Phi$

begin:

$\neg M$ についてこれを D リストの一部として持つ規則 $\rightarrow R_{Rela}$;

それに対応した前提条件 $\rightarrow P$;

M の P と矛盾する要素 $\rightarrow \Delta \neg M$;

$\Delta \neg M \rightarrow \neg M$;

end.

このようなアルゴリズムで簡単に関連規則が求められる。しかも関連規則の導出は部分的な計算ですむため, 関連規則の導出に要する時間は, プランニングにおける規則系列の導出に要する時間と比べればはるかに少ない。又, プランニングする時には, 関連規則を優先に選ぶことによりムダな探索をさけることができるので, 短い時間で最適プランを導出することも可能になりシステムの効率を高めることができる。