

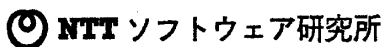
プランニングデータの解析を用いたプラン一般化手法

2M-6

伊藤智子

内藤昭三

伊藤正樹



1 はじめに

本稿では、数学的帰納法を用いてプランニングデータを解析し、一般化プランを作成する方法を提案する。次に、一般化プランに含まれるオペレータ列を優先的に探索し、プランニングの効率化を図るプランナーを提案する。最後に、縦型探索のプランナーと一般化プランを用いたプランナーで、評価実験を行い、一般化プランを用いたプランナーのプランニングコスト(探索時間)の優位性を示す。

2 一般化プラン作成

数学的帰納法を用いて一般化プランを作成する。数学的帰納法を用いるためには、

- 帰納変数の発見
- 帰納変数に対する操作方法の発見

が必要である。帰納変数として、要素の順序グラフを作成する。帰納変数に対する操作方法として埋め込みパターンを作成する。以下にアルゴリズムを示す。

2.1 リテラルの順序グラフ作成アルゴリズム

step1: オペレータの順序列で表されたプランから状態遷移図を作成する。

step2: 状態遷移図のゴール状態に含まれるリテラル(ゴールリテラル)を取り出す。

step3: ゴールリテラルが何番目に達成されるかを調べて、達成順にならべる。これをリテラルの順序グラフという。

2.2 要素の順序グラフ作成アルゴリズム

step4: リテラルの順序グラフから着目する述語を取り出して部分グラフを作成する。

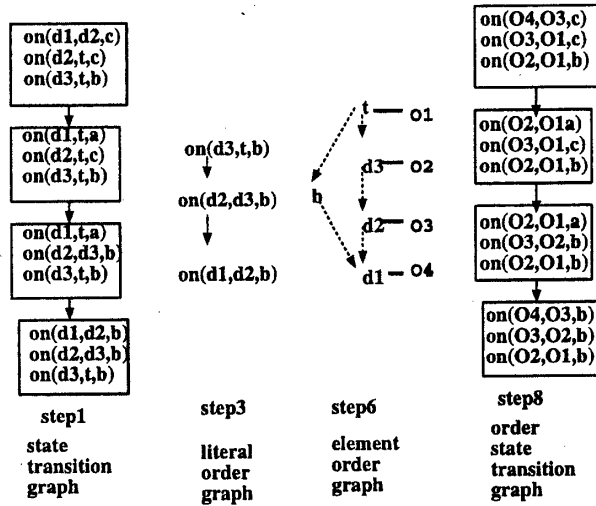
step5: step4で作成されたグラフから述語の引数のみを取り出す。

step6: 順序づけられたすべてのリテラル対中のすべての要素対に対して、リテラルの順序に基づく順序づけを行なう。これを要素の順序グラフと呼ぶ。

2.3 順序プラン、順序状態遷移図作成アルゴリズム

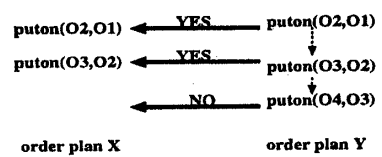
step7: プランおよび状態遷移図の中で、要素の順序グラフに含まれる要素を見つけ出す。

step8: その要素が要素の順序グラフの中で何番目に現れるかを調べて、1番目に現れたら、O1(=Order 1)に、I番目に現れたらOI(Order I)に要素の書き換えを行う。書き換えたプランを順序プラン、書き換えた状態遷移図を順序状態遷移図と呼ぶ。



2.4 埋め込みパターンの作成

要素順序数 N 個のプランと要素順序数 N-1 個のプランの対応関係を調べる。埋め込みパターンは順序要素 N 個のプランに対する順序要素 N-1 個のプランの埋め込み関係を表している。埋め込む先があれば Yes をなければ No を返す。



上記の例題の埋め込みパターンは [Yes, Yes, No] となる。また、order plan Y は order plan X と未知のオペレータを結合したもので表されることがわかる。

2.5 一般化プラン作成

一般化プランは、[順序プラン、順序状態遷移図、埋め込みパターン]の3つ組で表現される。3個のブロックを積み上げタワーを作るプランを例にとり、以下に示す。

プラン = [puton(い, う), puton(あ, い)],  
 状態遷移図 = [[clear(あ), clear(い), clear(う)],  
 [clear(あ), clear(い), on(い, う)],  
 [clear(あ), on(あ, い), on(い, う)]]  
 3個のブロック(順序要素は3個)に対する一般化プランは以下の通りである。  
 順序プラン = [puton(O2,O1), puton(O3,O2)],  
 順序状態遷移図 = [[clear(O3), clear(O2), clear(O1)],  
 [clear(O3), clear(O2), on(O2,O1)],  
 [clear(O3), on(O3,O2), on(O2,O1)]]  
 埋め込みパターン = [Yes, No].

3 一般化プランを用いたプランニング

予想プランを作成し、予想プランに含まれるオペレータ列を優先的に探索してプランニングの効率化を図る。プランニングアルゴリズムを以下に示す。

A Method for Generalizing a Plan Based on Analysis of Plan.  
 Tomoko ITO, Shozo NAITO and Masaki ITO.  
 NTT Software Laboratories

[プランニングアルゴリズム]

- step1: 入力として、求めたいプランの初期状態とゴール状態が与えられる。
- step2: 一般化プランは、[順序プラン、順序状態遷移図、埋め込みパターン]の3つ組で表現されている。順序状態遷移図の一番最初の状態と求めたいプランの初期状態を、順序状態遷移図の一番最後の状態と求めたいプランのゴール状態のユニフィケーションを行う。
- step3: step2で見つけたユニフィケーション結果を順序プランに適用する。
- step4: ユニフィケーションを行った後の順序プランに埋め込みパターンを適用して予想プランを作成する。
- step5: プランナーで、予想プランに含まれるオペレータ列を優先的に探索して、プランニングを行う。
- step6: step5の探索で適切なオペレータが見つからなかった場合は、単純な縦型探索のプランナーで適切なオペレータを探す。step5に戻る。

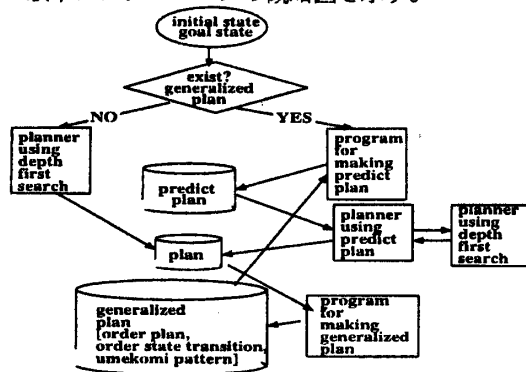
以下に順序要素3個の一般化プランを使ってプランニングを行う場合を示す。求めたい初期状態とゴール状態が以下の通りに与えられたとする。

初期状態 = [clear(a), clear(b), clear(c), clear(d)]  
 ゴール状態 = [clear(a), on(a,b), on(b,c), on(c,d)]  
 この入力と一般化プラン(2.5章11行目から15行目)とのマッチングを行うことにより以下の2つの予想プランが作成される。

O1 = c, O2 = b, O3 = a,  
 予想プラン1 = [puton(b,c), puton(a,b)]  
 O1 = d, O2 = c, O3 = b,  
 予想プラン2 = [puton(c,d), puton(b,c)]

次に、予想プランに含まれるオペレータ列 [puton(b,c), puton(a,b)] と [puton(c,d), puton(b,c)] を優先的に探索して、プランニングの効率化を図る。予想プランで予測できなかったオペレータに対しては、単純な縦型探索のプランナーを用いてオペレータを探索する。複数の予想プランがある場合はすべての予想プランに対して探索を行う。

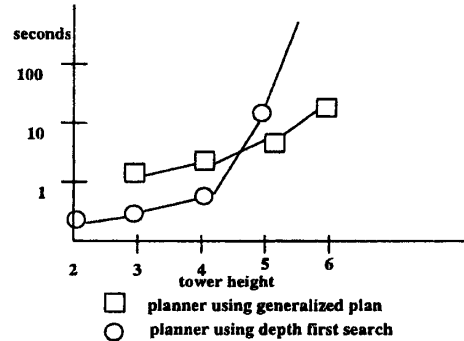
以下にプランニングの概略図を示す。



4 評価実験

上記の方法による一般化プランを用いたプランナーと単純な縦型探索によるプランナーとの評価実験を行った。N個のブロックからタワーを作る問題を評価実験の対象とした。プログラムはprologで作成し、ワークステーションSS20上で実験を行なった。3個のブロックを積むプランでは、一般化プラン作成、予想プラン

作成などのコストが予想プラン用いることによる探索コスト削減より大きいため、縦型探索のプランニングの方が早い。それに対し、ブロック数が5個以上になると一般化プランを用いたプランニングの方が縦型探索のプランニングより効率的であることがわかる。本評価実験で、ブロック数(順序要素数)が増えた時には一般化プラン作成がプランニングコスト(探索時間)の削減に役立つことが示された。



5 考察

縦型探索のプランナーと予想プランを用いたプランナーの比較を行う。

[縦型探索のプランナー]

長所: (理論的には)正しいプランが必ず作成される。  
 短所: 問題が複雑になると、計算量が爆発してしまい、コンピュータの資源の制約により解けなくなる。

[予想プランを用いたプランナー]

長所: 求めたいプランの初期状態とゴール状態と、一般化プランとのマッチングが主な処理のため、計算量は比較的少ない。

短所: 一般化プランと求めたいプランの初期状態とゴール状態を入力として求められる予想プランから、正しいプランが作成されるとは限らない。そのため、予想プランを検証し、不備な所(予想できなかった部分プラン)に対して、縦型探索のプランナーを使ってプランを作成する必要がある。

6 今後の課題

複数の予想プランが作成された時、本稿では、すべての予想プランに対してプランニングを続行したが、未知オペレータが初期状態に近い予想プランより、ゴール状態に近い予想プランの方が、プランニングの効率に優れていると考えられる。今後は、複数の予想プランの選択方法について検討を行っていく予定である。最後に、日頃御指導頂く知的ソフトウェア研究グループの皆様に深謝します。

参考文献

- 1) I. Bostrom: "Generalizing the Order of Goals as an Approach to Generalizing Number", 7th Conference of Machine Learning, 1991, pp. 260-267.
- 2) 伊藤(智)、鈴木、伊藤(正): "非線形的なプランニングにおける一般化手法の提案", 47回情報大全, 2p-10, 1993
- 3) 伊藤(智)、鈴木、伊藤(正): "順序構造を用いたプラン一般化手法について", 48回情報大全, 2N-1, 1994
- 4) 伊藤(智)、鈴木、内藤、伊藤(正): "埋め込み操作を用いたプラン一般化手法", 信学会研究会, AI94-23 32, 1994/7
- 5) 伊藤(智)、内藤、伊藤(正): "埋め込み操作を用いたプラン一般化手法とその評価", 49回情報大全, 3J-4, 1994
- 6) 後藤 滋樹: "Formal LISPにおけるトレースの標準形とその応用", コンピュータソフトウェア, Vol.7, No.2, (1990)