

高並列推論エンジンPIEにおける

処理の効率化について

4P-10

相田 仁 , 後藤 厚宏 , 田中 英彦 , 元岡 達

(東京大学 工学部)

1. はじめに

高並列推論エンジンPIEでは、並列アクティビティ間の独立性が高いため、きわめて高い並列性を取り出し得る。しかし、UP→MM、MM→UPのゴールフレームの転送ストラテジによっては特定のUPやMMにゴールフレームが集中し、実質並列度が上がらない可能性がある。また、PIEにおける並列アクティビティの大部分は、論理プログラムにおける選択枝の失敗の確認を行なっていることに相当するので、これらの選択枝は少なくできるものならば、少ない方がよい。

本発表では、PIEにおいて有効な並列処理を実現するための効率化手法について、時間を考慮した新しいシミュレータによるシミュレーション結果を報告する。

2. 効率化手法

PIEにおいて処理を効率化するための手法には、以下のようなものが考えられる[1]。

(1) 単一化の負荷の一部をDMに肩替りさせ、UPのスループットを向上させる。

PIEでは各UPにおいて1つのゴールリテラルとそれに対応するすべての定義節との間で単一化が行なわれる。ここで、UPにおける単一化のうちで失敗する場合に着目すると、簡単な照合操作により判定できる場合が多い。

そこでUPからDMに対し、リテラルの引数に関する情報を送り、あらかじめDMにおいて簡単な照合操作を行ない、単一化可能と判定されたものだけUPに送ることにすれば、転送のオーバーヘッドを減少させることができる。

(2) MM内のゴールフレームのうちで最適なものを選んでUPで処理する。

解をただ一つ得ればよい環境では、ゴールプール内のゴールフレームのうちで最も解に近いものから処理を進めるのがよい。また、すべての解を求めたい場合には、失敗に近いものから先に処理してしまい、ゴールプール内のゴールフレームをできるだけ少なく保つことも有効である。

(3) ゴールプール内のゴールフレームの中で、単一化の実行なしに失敗に到ると判定可能なものを消去する。

逐次型Prologにおける知的後戻りと同様な手法をPIEにおける効率化に役立てることができる。

(4) ゴールフレーム中のゴールリテラルのうちで最適なものを選んで単一化を行なう。

AND関係にあるゴールリテラルのうち、どれから単一化を行なうかによって探索木の形状が変化する。探索木の形状は異っても結果として得られる解の個数は同じであるから、探索木の大きさができるだけ小さくなるようにゴールリテラルの単一化順序を選ぶのがよい。

3. シミュレーションモデル

今回はUP/DMとAC/MMが一体化したモデルについてシミュレーションを行なった[2]。1つのUP-MMユニット内でのゴールフレームの転送遅延は0にしている。導出の各ステップに対応するシミュレーションクロック数を表1に示す(図1参照)。

今回のシミュレーションにおいては

$$t1=t2=t3=20, t4=t5=1$$

とした。

表1. シミュレーションクロック数

定義節絞り込みの準備	t1
定義節の転送および単一化	t2
組込み述語の実行	t3
縮退および新ゴールフレームの作成	t4 × n
ネットワーク遅延	t5 × n

(但しn=導出ゴールフレームのセル数)

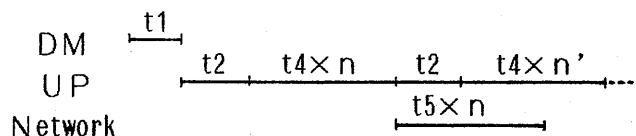


図1. 導出のステップ

4. シミュレーション結果

2. で述べた各種の効率化手法のうち、(1)の定義節の絞り込み、(2)のゴールフレーム分配、(3)のゴールフレームの選択について、いくつかのストラテジのシミュレーションを行なった。シミュレーションに用いた例題は6-Queens および数式簡単化Equiv2 [1, 3]である。

(1) 定義節の絞り込み

定義節の絞り込みによる実行時間の変化を表2に示す。数式簡単化のように同一述語名に関する定義節の数が多い場合には絞り込みの効果が大きく現れる。これに対し6-Queens では同一述語に関する定義節は高々2つであり、絞り込みの準備に要する時間がオーバーヘッドとなって、かえって性能が低下している。

(2) ゴールフレームの分配ならびにゴールフレームの選択

UPで導出したゴールフレームをどのMMに送るかのストラテジは次の3種をとりあげた。

- (A) すべてrandomに宛先を選ぶ
- (B) あるゴールフレームから導出されるゴールフレームの内最初のを自分に戻し、それ以外は他のMMにrandomに送る
- (C) 自分のMMが空のときだけ自分に戻し、それ以外は他のMMにrandomに送る

またMM内のゴールフレームのうちから次に処理するゴールフレームを取り出すストラテジとしては次の4つをとりあげた。

- (a) FIFO
- (b) 探索木上で深いものほど優先
- (c) 探索木上で浅いものほど優先
- (d) ゴールフレーム内のゴールリテラルのうち、第一レベルのものが少ないものほど優先 [1]

これらの分配ストラテジと選択ストラテジの組み合わせについてシミュレーションを行なった結果を表3に示す。

6-Queens の場合、ゴールフレームの分配ストラテジに関してはC, B, Aの順に所要時間が少なく、またゴールフレームの選択に関しては、bの深さ優先が最初の解を見つけるまでの時間が短く、cの広さ優先がすべての解を探索し終るまでの時間が短い。これに対しEquiv2においてはあまり統一した傾向は見られない。

5. おわりに

高並列推論エンジンPIEにおいて、実行時の動的な判断により処理を効率化する手法に関し、これまでにシミュレーションにより得られた結果を示した。

[参考文献]

1. 後藤 他：高並列推論エンジンPIEにおける並列処理の効率化手法について、通信学会技術報告EC83-9, 1983
2. 丸山 他：高並列推論エンジンPIEにおけるアクティビティ制御機構のシミュレーション、本大会予稿集4P-11, 1983
3. 相田 他：並列PROLOGシステムParalogの性能測定、24回情報処理全国大会5D-5, 1982.

Table 2. Effect of Filtering of Definitions (random / FIFO)

filtering method	6-QUEENS (30 UP's)		EQUIV2 (10 UP's)	
	first solution*	total execution	first solution	total execution
no filter	23191	25101	17652	29878
level 0	22567	24307	13334	21977
level 1	23249	25337	9307	12385

(in simulation clock)

* first of best solutions

Table 3. Effect of Distribution and Selection Strategy (filter level 1)

distribution / selection	6-QUEENS (30 UP's)		EQUIV2 (10 UP's)	
	first solution*	total execution	first solution	total execution
A / a	23428	24784	8079	12150
A / b	20913	29094	7526	13127
A / c	23528	24294	9689	12847
A / d	20929	25179	7553	12573
B / a	17112	18779	9213	14261
B / b	16757	20719	7942	15479
B / c	17444	18351	8301	12817
B / d	17720	19824	8198	13433
C / a	17018	19064	7906	12127
C / b	16422	19237	8577	14464
C / c	17129	17916	8646	12394
C / d	17389	18118	8349	11558

(in simulation clock)

* first of best solutions