

構造メモリ試作ハードウェアを用いた
PIEの全体性能向上の評価

猪股宏文, 平田圭二, 田中英彦
(東京大学 工学部)

4P-2

1. はじめに

現在、我々は高並列エンジンPIE[1]の構造メモリ試作ハードウェアの開発を進めている[2,3]。PIEの構造メモリ(SM)とは、構造データ共有方式をPIEに導入するためのグランドインスタンスを格納する共有メモリである。構造メモリ試作ハードウェアは、構造メモリと推論ユニット(IU)とホストコンピュータの3ブロックからなる実験システムとして構成されている。以下では、構造メモリによるPIEの性能向上の評価に必要な処理について説明し、簡単な例題でPIEの性能向上を考察する。以降、200nSECを単位としたステップ数を時間の単位とする。

2. 各処理の処理速度

PIEは、ゴール書き換えモデルによってOR並列に論理型プログラミング言語の実行を行う。この実行は、(1)ユニフィケーション(単一化)と(2)リダクション(縮退)と(3)ゴール節等のデータの転送の3つの処理を1周期とする推論サイクルの繰り返しによって行われる。それぞれ3つの処理で実行されるSMに対するサブコマンドについて説明する。

2.1. Lazy Fetch(LF)

LFの高速化はPIEの性能向上に必要なである。推論ユニット内のデータを処理する場合は従来と同等変わるところはないが、構造メモリ内にあるデータを処理する場合はLFで推論ユニットに必要なデータを持って来なければならぬために従来以上の処理時間を必要とする。

現在の制御用マイクロプログラムから換算すると1ノードのリストデータやベクトデータをLFするのに費されるステップ数は表1にまとめる。表1でWAITING TIMEはSMをアクセスするためにLazy Fetch Network(LFN)の使用権をLFNのアービタから得るのに費される時間である。従って、IUが複数台あったときに他のIUがSMをアクセスしていれば、それが終了しLFNを解放するまで待たなければならない。Lazy Fetch Buffer Table(LFBT:連想メモリで実装)に空領域が無くなると、LFBTに空

	CACHE HIT	CACHE NO HIT
LIST	10	42+WAITING TIME
VECT	10	39+4n+WAITING TIME (n: SIZE OF DATA)

表1. Lazy Fetchに費されるステップ数

領域をつくるためにWAITING TIMEが増加する。

2.2. 参照カウント(RC)命令の処理

IU内でRC命令が発生した場合、IU内のReference Count Buffer(RCB)に構造データ毎のバッファリングが行われる。これに費されるステップ数は、表2に示めす。

	NUM. OF STEPS
EXISTING IN RCB	5
NON-EXISTING IN RCB	6

表2. 参照カウント命令の処理に費されるステップ数

これは、SMに対するサブコマンドが上のステップ数より短い間隔で発生した場合、RC命令の処理のためにIUの処理が待たされる。表1のEXISTING IN RCBはRCBに既にバッファリングされているRC命令に対してそのカウンタの値を更新する処理を示し、NON-EXISTING IN RCBはRCBにバッファリングされていないRC命令をRCBにバッファリングする処理を示している。また、この処理はIUの内部のみで行われる。

2.3. Ground Instance(GI)の格納

IU内では、ゴール節から切り分けられたGIがIU内のGround Instance Buffer(GIB)に一旦貯えられ、SMへ送出するコマンドパケットの形にして

Command Output Buffer(COB)にバッファリングする。また、この処理ではGIの切り分けのときにSMの空アドレス(EA)をEmpty Address Buffer(EAB)から取り出して、EABにバッファリングされているEAが少なくなるとCOBにEAの要求命令をバッファリングする。表3にその処理に費されるステップ数を示す。

	NUM. OF STEPS
LIST	13
VECT	$14 + 2n$ (n : SIZE OF DATA)

表3. Ground Instance の格納に費されるステップ数

これは、IUが切り分けたGIをコマンドパケットの形にしてCOBにバッファリングする時間を示している。もし、IUにバッファリングされているEAが少なくなるとEA要求命令をCOBにバッファリングするため4から5ステップが表3の値に加算される。SMへのGIの格納はCOBにバッファリングされているコマンドパケットがSMに送出される時である。

2.4. コマンドパケットの送出

IUは推論サイクルの終了時にCOB内のコマンドパケットをSMへ送出する。コマンドパケットには、GIの格納命令とEAの要求命令がある。EAの要求命令は含まれない場合もある。もしEAの要求命令が含まれていればコマンドパケット送出終了後に続いてSMからのEAの受け取りの処理が行われる。この処理に費されるステップ数を表4に示す。

	NUM. OF STEPS
SENDING COMMAND PACKETS TO SM	$4 + n + \text{WAITING TIME}$ n : NUM. OF CELLS IN COB
RECEIVING EAs FROM SM	$9 + 2m + \text{WAITING TIME}$ m : NUM. OF EAs

表4. IUのコマンドパケット送出に伴う処理のステップ数

SMでは、このコマンドパケットをバッファで受け逐次処理し、1ノードのリストデータの格納に6step、ベクタデータの格納には $(14+n)$ stepが費され、 $m=30$ でEAのIUへの送出は40step費される[2]。

3. 全体性能の評価について

現在、構造メモリ試作ハードウェアはその一部をまだデバッグしている段階である。しかし、推論ユニットのシミュレータ[3]によって構造メモリの処理のステップ数を零としたときのデータが測定できるので、簡単な以下の例題に対して考察する。表5に全解の探索のときのシミュレータの実行結果を示す。

定義節:

append(X, [], X).

append([X|Y], Z, [X|YZ]) :- append(Y, Z, YZ),

ゴール節: (N = 0, 1, 2, 3, ...)

?- append(X, Y, [1, 2, ..., N]), print(X, Y) ;

N	REDUCTION RATIO OF EXECUTION STEPS(%)
NIL	0 (182step)
10	11 (7591step)
100	33 (434281step)

表5. 構造メモリによるIUの処理のスピードアップ率

表5のカッコで囲まれた値はSMが無いときの実行ステップ数である。LFの回数とGIの格納の回数はそれぞれ $N/2$ を超えない最大の整数として与えられる。概算で $(55 + \text{WAITING TIME}) \cdot N/2$ 程度のステップ数のスピードダウンが起きる。これはWAITING TIMEが数ステップのオーダーならば、全実行時間の数パーセント程度である。従って、本例題のような問題でも大きな構造データを処理する場合は数十パーセント程度のスピードアップが期待できる。

4. おわりに

構造メモリによるIUのスピードアップは、縮退と、ゴール節のコピー操作のそれぞれの実行時間の短縮によっている。これらの処理はゴール節のサイズが小さくなると、その実行時間も短くなる。LFによるアクセス頻度が少なくサイズの大きな構造データがあると大きなスピードアップが期待できる。

現在、構造メモリ試作ハードウェアは、まもなく終了する予定である。デバッグの終了後、より詳細なデータを取る予定である。

参考文献

[1] Moto-oka, T., Tanaka, H., et al., "The Architecture of A Parallel Inference Engine -PIE-", FGCS'84, ICF, 1984
 [2] 平田他, "PIEの構造メモリ試作ハードウェアのマイクロプログラム", 第33回情報全大, 5B-2, 1986.
 [3] 平田他, "PIEの構造メモリ試作ハードウェアの推論ユニット側ソフトウェア", 第33回情報全大, 5B-3, 1986.