

PIMの並列キャッシュ

ー共有バスの一構成法ー

3C-2

中川貴之、松本明、後藤厚宏

(財)新世代コンピュータ技術開発機構

1.はじめに

中期の並列推論マシンPIM(Parallel Inference Machine)のハードウェアは、図1に示すように共有メモリを介した密結合マルチプロセッサと、マルチPSIで導入している疎結合マルチプロセッサの二階層の構成をとる¹⁾。クラスタと呼ぶ密結合マルチプロセッサではPIMの対象言語であるKL1(Kernel Language/1)と、その処理系が有する参照局所性を活用するために並列キャッシュの導入を検討している。本発表では、クラスタ内処理におけるシミュレーション(CLSIM)結果の解析をもとに、共有バスの構成法の比較を試みる。

2.並列キャッシュにおける共有バスプロトコル

図1のような共有バス結合に基づく並列プロセッサにおいては、転送時間が一律に短いために使い勝手が良い反面、並列効果の上限がバストラフィックによって規定される短所がある。そこで、バストラフィックを削減する、さまざまな並列キャッシュプロトコルが提案されている²⁾。これらの並列キャッシュプロトコルの根本理念は“プロセッサ間に共有されないデータ”の状態を動的に管理する、バススヌープに体现される。すなわち真に必要なキャッシュ間/キャッシュ-メモリ間相互作用のみに共有バスを使うが、バスを共有することによる接続台数の頭打ちは本質的に回避できない。

プログラムを実行した場合のプロセッサからのメモリアクセス要求はスワップイン/アウトが同時に起きる。このことを考慮すると、スワップインのメモリアクセス中にスワップアウトデータを送る、以下のようなバストラランザクションにまとめることができる。

(1)バストラフィック発生要因

```

swapin(si)
swapin-with-swapout(siso)
cachexfer(cx)
cachexfer-with-swapout(cxso)
swapout(so)
invalidation(inv)

```

スワップアウトのみのバストラフィックは前回発表したDirect-write(dw)命令により発生する。これらの特性は以下のパラメタに還元できる。

(2)動的アクセス特性

```

acc : access-frequency(i.e.read+write)
hit : 1.00 - (si + siso + cx + cxso) / acc
common__access : (cx + cxso) / (si + siso + cx + cxso)
common__write : inv / (write × hit)
modify : so / (dw / BLK)

```

共有バスの使用サイクル数は、コマンドの転送サイクルを2サイクル、共有メモリのアクセスターンアラウンドタイムをTaccサイクル、キャッシュのブロック長をBLK、転送コストをXcostとして、以下のように定式化できる。

(3)バストラフィックコスト

```

swapin(si)...BLK + Tacc + 2
swapin-with-swapout(siso)
...BLK + max(BLK, Tacc) + 2
cachexfer(cx)...BLK + Xcost + 2
cachexfer-with-swapout(cxso)
...BLK + max(BLK, Xcost) + 2
swapout(so)...BLK + 2
invalidation(inv)...1

```

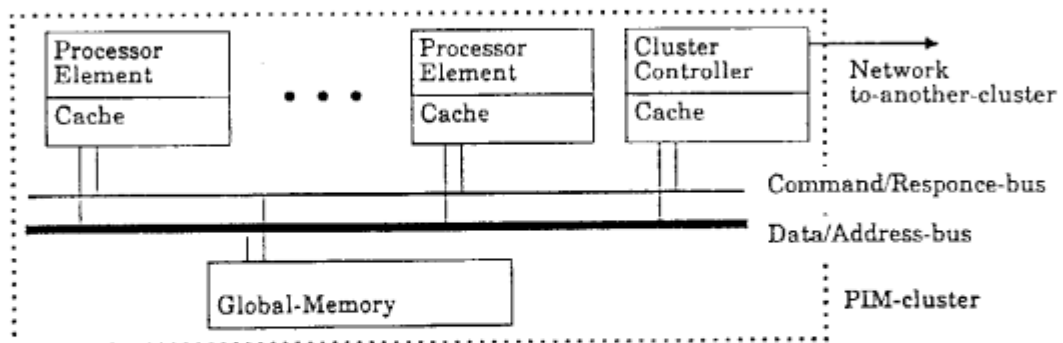


図1 共有バスモデル

Parallel Cache for PIM- On Common Bus Configuration -

Takayuki NAKAGAWA, Akira MATSUMOTO, Atsuhiko GOTO

Institute for New Generation Computer Technology (ICOT)

3. 共有バスのスループット改善

我々が作成したクラスタの実験処理系において BUP(Bottom Up Parser)のプログラムを8台のプロセッサからなるクラスタで実行した場合の各プロセッサからのメモリアクセスパターンによるキャッシュの動作をシミュレートした。これによると、BUPでは36Kリダクションを570KステップのKL1-b命令(KL1のマシン語)で処理するに際し、1.3Mアクセスに259Kbus-cycleを要する。

尚、Tacc=4, Xcost=0, キャッシュサイズは4Kword/peで、4word×256column×4setの簡易LRU制御を仮定している。

動的アクセス特性としては、hit=98%, comm_a=86%, comm_w=2%, modify=71%と、キャッシュが有効である反面、共有率が高く複数プロセッサによる処理の分割損があること、従って高速なキャッシュ間転送が必須であることがわかる。バスサイクルタイムを100nsと仮定するとプロセッサあたりの性能目標200Krpsに対し、名目バス使用率は115%となりバス競合による待ち合わせ時間が無視出来ない。そこでバススループットを向上する為、バスサイクルタイムを50nsとし、Tacc=8、Xcost=1、キャッシュタグは1サイクルに1回しかアクセス出来ないとの仮定をたてた。この結果バス使用時間は44%減るが、名目バス使用率は65%あるので、更に以下の3つの案を評価した。

- (1)バス幅を2ワードにする
- (2)用途別バスにする
- (3)インタリーブバスにする

また、先のバストラフィックコストの算定式には以下の仮定をしている。

- (a)スワップインのメモリアクセスタイムはメモリコントローラの簡易の為、バスビジーとする。
- (b)スワップアウトのメモリアクセスタイムはバスビジータイムに含まない。

すなわち、バスサイクルタイム100nsでは、仮定(a)による損害は3%程度であったが、バスサイクルタイム50nsでは17%の損害がある。そこで、バスを解放するが、キャッシュタグの更新に最低2サイクルを要すると考え10%の削減を見積もった。この場合の名目バス使用率は58%である。なお、仮定(b)のメモリビジーによるバストラフィック増は最悪でも2%未満とみている。

評価の結果以下の効果が明らかになった。

(案1)バストラフィックを36%削減する。

(案2)バストラフィックを27%削減する。

(案3)バストラフィックを50%削減する。

インタリーブバスはキャッシュタグのスループットを増やす必要があるが、キャッシュタグのビジー率が200Krps/peで48%であるので、むしろ望ましい。また、クラスタのプロセッサ台数を増やす上で拡張性があり、ピン効率の上からも確実なスループット増が期待できるので有望視している。

4. 結論

本発表ではクラスタ内処理におけるシミュレーション(CLSIM)結果の解析をもとに、共有バスの構成法の比較を試みた。今後は更に並列度をあげる検討を進める予定である。

<参考文献>

- (1)中川、松本、清水、他、"並列推論マシンPIM", 情報処理学会第34回全国大会予稿集2P-1~9
- (2)P.Bitars, A.M.Despain, "Multiprocessor cache synchronization", 13th Int.Symp.on Comp.Arch.
- (3)L.Rudolph, Z.Segall, "Dynamic decentralized cache schemes for MIMD parallel processors", 11th Int.Symp.on Comp.Arch., June 1984

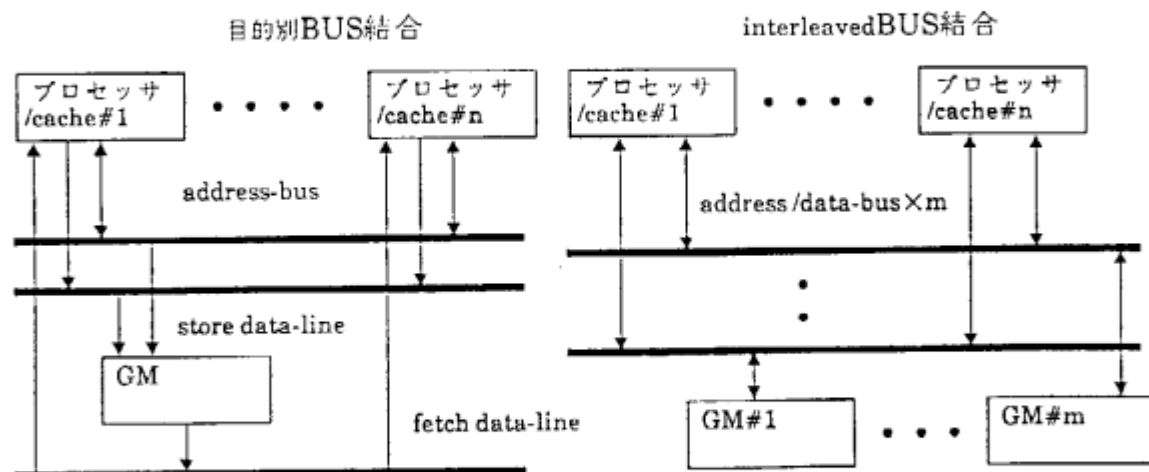


図2 共有バス構成