

## 2H-9

並列推論マシンPIM/iの  
メモリスステムの評価

- 並列キャッシュ -

大原輝彦 武田浩一 佐藤正樹  
沖電気工業(株)総合システム研究所 (財)新世代コンピュータ技術開発機構

## 1 概要

筆者らは、第5世代コンピュータ・プロジェクトの一環として、並列推論マシンPIM/iの研究開発[1]を行っている。PIM/iは、8台のプロセッシングエレメントから成る共有メモリ型密結合マルチプロセッサシステムである。共有メモリ型密結合マルチプロセッサは、システムの構築が容易な反面、共有メモリのアクセスに使用する共有バスがボトルネックとなる問題がある。これを解決するため、筆者らは、各プロセッシングエレメントに局所メモリとキャッシュメモリを実装し、共有バスへのアクセス頻度を軽減するというアプローチ[1]を採用した。本稿では、PIM/iに実装した並列キャッシュメモリのシミュレータ上での評価について述べる。

## 2 並列キャッシュ

共有バスのトラフィック低減と共有メモリの参照を高速化する目的で、プロセッシングエレメント毎に並列キャッシュを実装した。以下に、PIM/iにおける実現手法を示す。

## 2.1 ライトバック型キャッシュ

キャッシュエントリのスワップアウトが必要になるまで共有メモリに書き戻しを行わない方式を採用した。PIM/iでは、キャッシュメモリに高速SRAMを使用し、共有メモリには大容量DRAMを使用している。動作速度の点からなるべく共有メモリをアクセスしなのが好ましいと考えた。

## 2.2 更新型一貫性制御プロトコル

キャッシュ間の一貫性制御のために更新型プロトコル[2]を採用した。この場合、キャッシュエントリの更新の際に、共有メモリに書き戻しを行うかどうかの選択枝があるが、共有メモリの動作速度の点から書き戻しを行わない方式とした。更新型プロトコルでは、共有状態のキャッシュエントリに書き込みを行うとバストラフィックが生ずる。高いヒット率を期待できるが、不必要なバストラフィックが生じる可能性もある。これを低減するために非共有(Exclusive)状態を導入し共有状態と区別する最適化を行った。

## 2.3 6状態キャッシュ

キャッシュエントリの制御を以下に示す6つの状態を用いて行う。

- ・ I (Invalid):  
エントリが無効であることを示す。

- ・ EOC (Exclusive Owner Clean):  
唯一のコピーであり、書き戻しの必要がないエントリである。
- ・ EOD (Exclusive Owner Dirty):  
唯一のコピーであるが、書き戻しの必要があるエントリである。
- ・ SUC (Shared Unowner Clean):  
共有されている可能性のあるエントリである。
- ・ SOC (Shared Owner Clean):  
共有されている可能性のあるエントリであり、キャッシュ間通信の転送元となる。
- ・ SOD (Shared Owner Dirty):  
共有されている可能性のあるエントリであり、キャッシュ間通信の転送元となる。また書き戻しの必要もある。

## 3 評価

キャッシュメモリの評価のためには、実行時間、コードサイズなどが十分に大きいプログラムを使用するのが一般的である。しかしPIM/iのコードキャッシュサイズは32K語と大きく、小さなプログラムはランタイムライブラリも含めてすべてオンキャッシュになる可能性が高い。実際、コードミスに伴うバストラフィックはほとんど発生しないという状況で並列キャッシュの評価を行った。

## 3.1 評価方法

PIM/iの開発環境である命令レベルシミュレータ[3]をベースに、キャッシュをシミュレートする機能を追加したシミュレータを作成し比較するという方法をとる。作成したシミュレータの概要を以下に示す。

## (1) 8ポートモデル・シミュレータ

共有メモリを8ポートメモリとみなし、共有メモリのアクセスに全く競合が発生しないと仮定したモデルをシミュレートする。

## (2) ライトバックモデル・シミュレータ

2節で述べたモデルをシミュレートする。

## (3) ライトスルーモデル・シミュレータ

キャッシュにライトヒットしたとき、同時に共有メモリと他のキャッシュの内容を更新するモデルをシミュレートする。

## 3.2 ベンチマークプログラム

ベンチマークプログラムには8クインーン問題を用いた。このプログラムをPIM/iのキャッシュプロトコルを考慮した処理系[4]上で実行する。ベンチマークプログラムの緒元を表1に示す。

表1 ベンチマークプログラム諸元

コードサイズ	7.7k語
リダクション数	39k回
メモリアクセス数	736k回

3.3 評価結果

ベンチマークプログラムをプロセッサの台数を変えて実行した結果を図1に示す。また、同様にプロセッサの台数を変えた場合のアイドルゴールのリダクション数を図2に示す。

この結果によると、8台のプロセッサを用いた場合の台数効果は、8ポートモデルでは、6.49倍、ライトバック型で5.93倍、ライトスルー型で3.00倍であった。8ポートモデルではメモリアクセスにおける競合が全く存在せず、理想的な場合と比較したオーバーヘッドはプロセッサ間通信と負荷分散によるものと考えられる。これに対し、キャッシュを持ったモデルではメモリアクセスにコストがかかる分、ゴールリダクションあたりの実行サイクル数が大きくなる。またアイドルゴールのリダクション数が増加が見られることから、負荷分散に影響を与えていることが考えられる。ライトバックモデルは、8ポートモデルと比べて最大1.15倍のオーバーヘッドがあることがわかる。

ライトバックモデルとライトスルーモデルを比べると、図2で明らかなように、前者のアイドルゴールのリダクション数は、8ポートモデルと比べて大きな差はないが、後者では、プロセッサ台数が8台のとき、アイドルゴールのリダクション数が増加し、台数効果が悪化している。この原因は、共有メモリのアクセスコストが高いために起こるバストラフィックの増加によるものと考えられる。その結果、負荷分散が影響を受け、アイドルゴールのリダクション数がライトバックモデルの2.5倍に増えている。また、プロセッサのアイドル率(パイプラインが空である割合)も約50%にも達していることがわかった。

ライトバックモデルの場合、更新に伴うトラフィックによるオーバーヘッドが存在すると考えられるが、8台のプロセッサでベンチマークを実行させた後、キャッシュエントリの共有ブロックの割合を調べてみると7.5%であった。さらに、更新に伴うバストラフィックを調べた結果、全体の約10%であった。

以上の結果から、共有メモリのアクセスコストが高い場合、ライトバックモデルはライトスルーモデルと比べて有効であることがわかった。また、バストラフィックの増加は負荷分散に影響を与える傾向が存在することがわかった。

4 おわりに

並列推論マシンPIM/iの並列キャッシュの構成と評価について述べた。現在は、小規模なベンチマークで評価を行っている段階であるが、今後さら

に、大規模なベンチマークを用いて評価を進めてゆく予定である。

謝辞 :

日頃助言をいただくICOT第1研究室、および沖電気のPIM担当諸氏に感謝する。

参考文献:

- [1]大原他: 並列推論マシンPIM/iの概要、情報処理学会第40回全国大会
- [2]J.Archibald, J.Baer, "Cache Coherence Protocols: Evaluation Using a Multiprocessor Simulation Model", ACM Trans. on Computer Systems, Vol4, No4, pp237-298, 1986
- [3]吉田他: 並列推論マシンPIM/iの開発支援環境-シミュレーター、情報処理学会第41回全国大会
- [4]久野他: 並列推論マシンPIM/iの処理系の概要、情報処理学会第41回全国大会

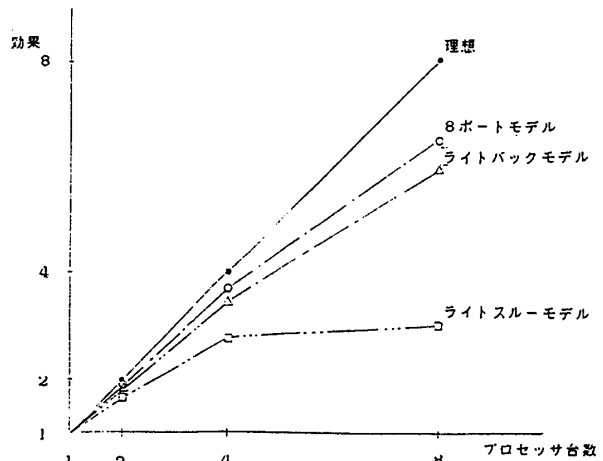


図1 台数効果

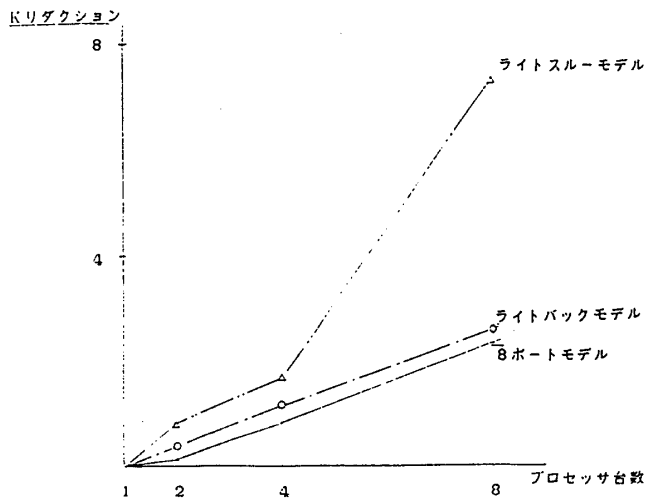


図2 アイドルゴールのリダクション数