

可変構造型並列計算機のキャッシュの単体性能評価

4 L - 6

岩田英次 森眞一郎 村上和彰 福田晃 富田眞治

(九州大学大学院総合理工学研究科)

1. はじめに

現在我々は、128台のPE(Processing Element)を128×128のクロスバー網で接続するマルチプロセッサ・システム「可変構造型並列計算機」を開発している^[1]。本システムでは、128台のPEそれぞれに容量4MBのローカルメモリおよび容量128KBの仮想アドレス・キャッシュを備えている。また、「ローカル/リモート・アーキテクチャ(分散共有メモリ構成)」を採用しているため、任意のPE群でメモリを共有する形態を採ることが可能となる。以上より、本システムは以下の二種の一貫性を保証する必要がある^[2]。

① “キャッシュ-キャッシュ間”コヒーレンス
(キャッシュ・コヒーレンス問題)

② “キャッシュ内”コヒーレンス(シノニム問題)

これらの一貫性を保証するため、本キャッシュでは仮想アドレスのみならず実アドレスによるアクセスをも可能としている。したがって、キャッシュは次の二つのディレクトリを備えたデュアル・ディレクトリ・キャッシュ(以下DDCと呼ぶ)構成となっている。

① Vタグ・アレイ: 仮想アドレス・タグ(Vタグと呼ぶ)等を保持するディレクトリ

② Rタグ・アレイ: 実アドレス・タグ(Rタグと呼ぶ)等を保持するディレクトリ

本稿では、DDCにおける“衝突現象”による性能低下の考察を行った後、確率的解析およびトレース駆動シミュレーションの二手法を用いて、DDCの単体性能の評価を行う。

2. DDCにおける衝突現象

DDCでは、キャッシュの大容量化のために、セット・アドレスにアドレス無変換部分(ページ内オフセット)のみならず変換部分をも含んでいる。したがって、複数の仮想セット・アドレスが同一の実セット・アドレスにマッピングされる可能性がある。さらにDDCにおいては、対応関係にある有効なVタグとRタグはダブルリンクしていなければならない(図1(a)参照)。このダブルリンクは、ライン・リプレース時に対応関係のあるVタグとRタグの間に張られる。これにより、ライン・リプレース時のリンクの張り替えの際に、“衝突現象”が発生する場合がある。この様子を図1の(b)に示す。

① まず、VタグaとRタグaの組が有効である。ここで、仮想アドレスbによりミスヒットを起こし、ライン・リプレース処理に入る。

② ライン・リプレースに伴い、Vタグ・アレイおよびRタグ・アレイに、VタグbおよびRタグbを登録しなければならない。このとき、Rタグbを登録すべきエンタリに、すでにRタグaが登録されていた。すなわち、RタグaとRタグbが衝突した。

③ Rタグbを登録するためには、Vタグa-Rタグa間のダブルリンクを解く必要がある。その結果、もともと有効であったVタグaが無効となる。つまり、Vタグaはライン・リプレースによって生じた“犠牲者(victim)”といえる。

④ ライン・リプレース処理が完了したとき、Vタグb-Rタグb間にダブルリンクが張られ、これらの組が有効となる。

衝突により、キャッシュの性能は低下する。衝突はRタグ・アレイの連想度のあるウェイ数以上にすると生じない。そのウェイ数は、キャッシュ容量とページ・サイズの比より求まる。本キャッシュの場合、動作モードに応じて、その比は16あるいは32となる。しかし、Rタグ・アレイの連想度を16ウェイあるいは32ウェイにすることは、ハードウェア・コストの爆発的な増大を招き、現実的でない。したがって、Rタグ・アレイの連想度の決定に際しては、衝突によるキャッシュの性能低下とハードウェア・コストとのトレードオフが問題となる。以下、Rタグ・アレイの連想度と衝突の頻度との因果関係を明らかにする。

3. 確率的解析

マルコフ連鎖を用いた確率的解析により衝突現象の影響を考察する。まず、以下の定義を行う。

E : キャッシュのエンタリ数

CF : キャッシュ容量とページ・サイズの比

n : Rタグ・アレイの連想度 ($n \leq CF$)

i : 有効なエンタリ数 ($1 \leq i \leq E$)

m_i : エンタリ数*i*の(Rタグ・アレイのない)キャッシュにおけるミスヒット率

n_i : 定常状態でDDCが状態*i*にいる確率

p_{ij} : 状態*i*から状態*j*への定常推移確率

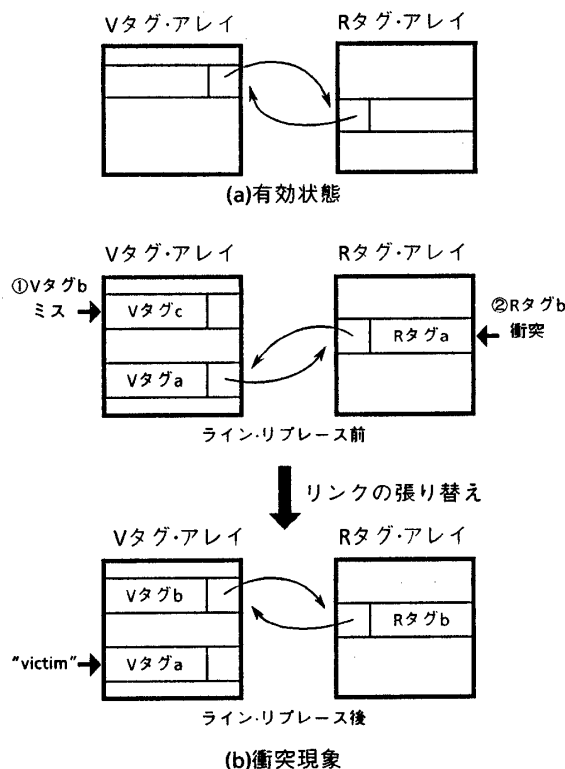


図1. Vタグ-Rタグ間リンク

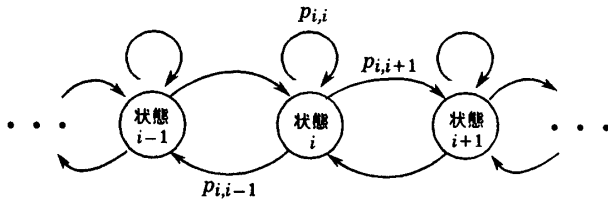


図2. DDCにおける状態遷移図

図2にDDCの状態推移の様子を示す。いまDDCが状態*i*にいるとすると、キャッシュミスによりDDCの状態は次のように推移する(図1(b)参照)。

- ① 状態*i*→状態*i+1*: 無効なVタグ*c*とRタグを有効なVタグ*b*とRタグ*b*でリプレースした。よって、有効なエントリ数が1個増える。
 - ② 状態*i*→状態*i-1*: 対応関係のない有効なVタグ*c*とRタグ*a*を有効なVタグ*b*とRタグ*b*でリプレースした。すなわち、Rタグ*a*とRタグ*b*が衝突した。このとき、当Rタグ*a*と対応関係にあるVタグ*a*が犠牲となり、有効なエントリ数が1個減る。
 - ③ 状態*i*→状態*i*: (I) 有効なVタグ*c*と無効なRタグ、(II) 対応関係にある有効なVタグ*c*とRタグ*c*、(III) 無効なVタグ*c*と有効なRタグ*a*、を有効なVタグ*b*とRタグ*b*でリプレースした。この場合、有効なエントリ数は不変である。ただし、(III)の場合、Rタグ*a*とRタグ*b*が衝突する。
- さて、定常推移確率 $p_{i,i+1}$ および $p_{i,i-1}$ は、次のようになる。

$$p_{i,i+1} = n_i \cdot m_i \cdot \left(\frac{E-i}{E}\right) \cdot \left\{1 - \left(\frac{i}{E}\right)^n\right\}$$

$$p_{i,i-1} = n_i \cdot m_i \cdot \left(\frac{i}{E}\right)^{n+1} \cdot \left(1 - \frac{n}{CF}\right)$$

ここで、定常状態においては、次のように $p_{i,i+1} = p_{i+1,i}$ が成り立つ。

$$n_i \cdot m_i \cdot \left(\frac{E-i}{E}\right) \cdot \left\{1 - \left(\frac{i}{E}\right)^n\right\} = n_{i+1} \cdot m_{i+1} \cdot \left(\frac{i+1}{E}\right)^{n+1} \cdot \left(1 - \frac{n}{CF}\right) \dots \textcircled{A}$$

($i = 1, 2, \dots, E-1$)

また、次の正規化条件が成り立つ。

$$\sum_{i=1}^E n_i = 1 \dots \textcircled{B}$$

$E-1$ 個の式④と式⑤より*E*個の n_i を求める。これより、以下のように、エントリ数*E*のDDCのミスヒット率*m*、および、使用エントリ数*u*を求める。

$$m = \sum_{i=1}^E m_i \cdot n_i$$

$$u = \sum_{i=1}^E i \cdot n_i$$

結果を表1に示す。ただし、エントリ数*E*は2048とし、*CF*は16とする。なお、 m_i の値として、後述するトレース駆動シミュレーションのRタグ・アレイなしの場合の結果を用いた。表1より、Rタグ・アレイの連想度を大きくすることにより、キャッシュの性能は向上すると考えられる。

4. トレース駆動シミュレーションによる解析
Sun-4 (SPARC) 上でトレースデータを採取し、トレース

表1. 確率的解析結果

	RTAなし	RTAあり		
		W=1	W=2	W=4
ミスヒット率 <i>m</i> (%)	9.316	9.477	9.414	9.369
キャッシュ使用効率 <i>u/E</i> (%)	100	49.12	63.13	78.08

RTA: Rタグ・アレイ

W: ウェイ数

駆動シミュレーションにより本キャッシュの単体性能を測定した。テストプログラムにはDhrystoneを用いた。表2にRタグ・アレイの連想度をパラメータとした測定結果を示す。前提条件は次の通りである。

- ① “キャッシューキャッシュ間”コヒーレンス保証の際の無効化は生じない。
- ② “キャッシュ内”コヒーレンス保証の際の無効化は生じない(シノニムは生じない)。
- ③ 仮想アドレスと実アドレスをある規則(セット・アドレスの変換部分とVタグの下部部分をスワップ)により対応づける。したがって、衝突は起こり得る。
- ④ ストア・スルー方式を採用するため、ストア・アクセスはすべてミスヒットとする。

表2より、ミスヒット率およびキャッシュ使用効率のいずれに対しても、Rタグ・アレイの連想度が大きく影響することがわかる。特に、連想度を1(ダイレクト・マッピング)としたときのヒット率低下は著しい。また、連想度が4の場合のキャッシュ使用効率には大きな向上が見られる。

5. おわりに

以上、確率的解析およびトレース駆動シミュレーションによる解析という2つの異なるアプローチにより、キャッシュの単体性能の解析を行った。これら2つの解析結果より明らかのように、DDCにおけるRタグ・アレイの連想度を大きくすることにより、キャッシュの単体性能の向上、さらにはキャッシュ・システム全体の性能向上が図れる。なお、ページング・スーパーバイザにより、仮想アドレスから実アドレスへのマッピングを衝突の頻度を減らすように行うことも可能であるが、ソフトウェアに対する制約となり好ましくない。よって、本キャッシュのRタグ・アレイの連想度は、ハードウェア・コストとのトレードオフの結果、4としている。

参考文献

[1] K. Murakami et al. : The Kyushu University Reconfigurable Parallel Processor - Design of Memory and Intercommunication Architectures - , Proc. 989 Int'l Conf. Supercomputing, pp. 351-360, June 1989.
[2] 岩田ほか: 可変構造型並列計算機のキャッシュ・システム, 情報処理学会計算機アーキテクチャ研究会資料, 89-ARC-79-3 (1989年11月)。

表2. シミュレーションによる解析結果

	RTAなし	RTAあり		
		W=1	W=2	W=4
メモリ参照回数		500,000		
命令フェッチ回数		381,068		
ロード回数		75,913		
ストア回数		43,019		
リプレース回数	3,134	10,870	3,500	3,333
キャッシュミス回数	46,582	53,907	46,624	46,599
衝突回数		9,602	2,188	1,421
ミスヒット率(%)	9.316	10.781	9.325	9.320
キャッシュ使用効率(%)	100	30.91	37.40	63.62

RTA: Rタグ・アレイ

W: ウェイ数