

Title	階層型キャッシュシステムにおける高効率なブロック配置法
Author(s)	許, 允碩
Citation	
Issue Date	2011-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/9630
Rights	
Description	Supervisor:田中清史, 情報科学研究科, 修士

階層型キャッシュシステムにおける高効率なブロック配置法

HUH, Younsuk(0910055)

北陸先端科学技術大学院大学 情報科学研究科

2011 年 2 月 8 日

キーワード: 階層型キャッシュ, ブロック配置, マルチコア.

1 背景

現在主流となっている多くのプロセッサは階層化されたキャッシュメモリの管理方法として Inclusion Property 方式が採用している. Inclusion Property 方式は上位キャッシュのブロックのコピーを下位キャッシュで持つため, プロセッサ間コヒーレンス処理が単純になる利点があるが [1], 同一のブロックのコピーが上位, 下位キャッシュで複数存在するためメモリ資源使用に無駄が発生する [2][3]. シングルコアプロセッサでは上位キャッシュと比較して下位キャッシュの容量が大きいためこのことは問題にならないが, マルチコアでは各コアが独立した上位キャッシュを持つ場合, それらの総容量は下位キャッシュに比べて無視できない. 一方, 他の方式として上位キャッシュのブロックのコピーを下位キャッシュで持たない Exclusion Property が存在する. Exclusive Property 方式では, 省メモリ資源の特性を持つ代わりに [2][3], コヒーレンス維持のための処理が複雑になりオーバーヘッドが大きくなる欠点がある [1].

本論文ではブロックを適切に配置することにより効率の良い資源使用を達成することと, コヒーレンス処理のオーバーヘッド問題を解決する方法を提案する. 又, 本論文で提案する手法の有効性を検証するために提案手法を適用したシミュレータを作成し, 評価を行う.

2 局所性に基づくブロック配置

本論文ではキャッシュのデータの参照をいくつかのパターンに分類し, 適切なキャッシュ階層に配置する手法を提案する. 長期的に高頻度で参照されるデータを含むブロックは, 上位キャッシュで置換された後も, 近い将来再度参照される可能性が高い. このようなブロックは上位キャッシュと下位キャッシュ両方に配置する. 長期的な再利用性があるが短期的には参照頻度が低い (すなわち, 参照間隔が大きい) データからなるブロックは, 最下位キャッシュにのみ配置することにより, 上位キャッシュ内の局所性の高いブロックを置

き換えることを防ぐ。ただし、ブロック内の空間的局所性に対処するために、当該ブロックを後述のバッファに格納する。短期的に高頻度で参照されるが長期的には再利用性が低いデータからなるブロックは、上位キャッシュにのみ配置することにより、下位キャッシュ内の再利用性のあるブロックを置き換えることを防ぐ。再利用性の無いデータからなるブロックは、上位、下位キャッシュに配置しないことにより、局所性の高いブロックを置き換えることを防ぐ。ただし、ブロック内の空間的局所性に対処するために、当該ブロックを後述のバッファに格納する。複数のプロセッサで共有されるデータは、一般的に下位キャッシュ間のスヌーピング機構により一貫性が維持される。Exclusion Property 方式では、上位キャッシュもスヌーピングの対象とする必要があるため、一貫性維持処理のオーバーヘッドが大きくなる。したがって、共有データを含むブロックは参照パターンによらず下位キャッシュに配置することにより、オーバーヘッドを削減する方針を採る。上位キャッシュに配置するか、あるいはバッファに配置するかは参照パターンに依存する。

本研究で提案するキャッシュシステムにはバッファが存在する。バッファは1次キャッシュと同列参照される位置に存在し、プログラムの要求する空間的な局所性のみを持つ参照列に対応する機構である。したがって、バッファのサイズは数ブロックほどの極小さいもので十分である。バッファは1次キャッシュと並列参照し、1次キャッシュと同等かそれ以上早く参照が完了するサイズを想定する。

3 シミュレーション

提案手法の有効性を検証するために、提案手法を実装したマルチコア（2コア）プロセッサシミュレータを作成した。作成したシミュレータはSPARC Architecture Version 8[4]の命令セットを使用する。シミュレータのキャッシュは命令・データ独立1次キャッシュと命令・データ共有の2次キャッシュの2段階層を持つキャッシュとデータバッファで構成される。ベンチマークプログラムはSPLASH-2[5]のカーネルプログラムを用いた。シミュレーションは2つのプログラムを同時に実行させた場合、単一のプログラムを並列計算させた場合に分けて行った。性能比較の対象は、従来のInclusion property/Exclusion property 機構を持つシミュレータでベンチマークプログラムを実行させた場合の実行時間と、提案手法のブロック配置機構を適用したシミュレータでベンチマークプログラムを実行させた場合の実行時間である。

4 まとめ

本研究では、階層型キャッシュにおけるInclusion Property方式のメモリ資源の浪費とExclusion Property方式のコヒーレンス処理のオーバーヘッドの問題に対して、キャッシュブロックの適切な配置により解決するキャッシュメモリシステムの提案を行った。また、

提案手法の検証を行うために、提案手法を適用したシミュレータを作成し、ベンチマークプログラムを実行させることにより評価を行った。

参考文献

- [1] J.-L. Baer and W.-H. Wang. On the inclusion properties for multi-level cache hierarchies. *SIGARCH Comput. Archit. News*, 16:73–80, May 1988.
- [2] Ying Zheng, B. T. Davis, and M. Jordan. Performance evaluation of exclusive cache hierarchies. In *Proceedings of the 2004 IEEE International Symposium on Performance Analysis of Systems and Software*, pages 89–96, Washington, DC, USA, 2004. IEEE Computer Society.
- [3] Manoj Franklin Mohamed M. Zahran, Kursad Albayraktaroglu. Non-inclusion property in multi-level caches revisited, June 2007.
- [4] *The SPARC Architecture Manual*, 1991,1992.
- [5] Stanford parallel applications for shared memory (splash-2).