

# 一种面向 CMP 的可变相联度混合 Cache 结构

晏沛湘, 杨先炬, 张民选

(国防科学技术大学计算机学院, 湖南长沙 410073)

**摘要:** 以 V-Way Cache 结构为原型, 提出一种面向 CMP 的可变相联度混合 Cache 结构 CMP-VH. CMP-VH 将最后一级片上 Cache 划分成一种优化的私有/共享结构, Tag 私有, 数据部分私有部分共享. 采用基于数据块的重用信息替换策略, 提供显式和隐式两种机制在核间对共享数据进行容量划分. 并行程序负载 SPLASH-2 的模拟实验结果表明, CMP-VH 具有比单一的私有/共享结构更好的整体性能.

**关键词:** 片上多核处理器; 混合 Cache 结构; Reuse 替换策略

**中图分类号:** TP302      **文献标识码:** A      **文章编号:** 0372-2112 (2011) 03-0656-04

## A CMP Oriented Variable-Way Hybrid Cache

YAN Pei-xiang, YANG Xian-ju, ZHANG Min-xuan

(College of Computer, National University of Defense Technology, Changsha, Hunan 410073, China)

**Abstract:** We propose a CMP oriented variable-way hybrid cache mechanism, named CMP-VH, based on V-Way Cache. CMP-VH turns the last level cache into an optimized private/shared organization. The tag array is private, while the data array is private and shared organized. In CMP-VH, we use reuse information for cache line replacement, and provide implicit and explicit partitioning mechanisms for shared data region. The simulation result of parallel workload SPLASH-2 shows that CMP-VH excelled the pure private or shared cache.

**Key words:** chip multiprocessors; hybrid cache organization; reuse replacement

## 1 引言

私有和共享结构是 CMP 中最后一级 Cache 结构设计的两种基本方案. 私有结构具有延迟优势, 共享结构具有容量优势, 单一的私有或者共享结构都不能实现最优性能. 本文目标是权衡延迟和容量, 设计一种优化的混合 Cache 结构.

现有 Cache 结构广泛采用 LRU 替换策略. 随着应用的日趋多样复杂化, LRU 替换策略与最优替换策略之间的差距越来越明显. Qureshi 等<sup>[1]</sup>提出 V-Way Cache 结构采用 Reuse 替换策略, 试图以较少的硬件开销实现全局替换. V-Way Cache 具有一定的应用自适应能力, 在单核环境下取得较好的效果.

本文以 V-Way Cache 结构为原型, 针对最后一级 Cache 层次, 提出一种面向 CMP 的可变相联度混合 Cache 结构 (CMP-Oriented Variable-way Hybrid cache, 简称为 CMP-VH). CMP-VH 在私有 V-Way Cache 的基础上扩充容量共享, Tag 区私有, 数据区私有/共享并存. 为了实现容量与需求的自适应, CMP-VH 使用扩展的 Reuse

替换策略 (Extended Reuse replacement, 简称为 E-Reuse 替换) 管理数据资源, 一方面基于重用频率进行替换, 另一方面提供隐式和显式两种机制在核间划分共享数据区. 模拟实验结果表明, CMP-VH 不仅兼具私有结构低延迟、共享结构低失效率的特点, 还能适应不同应用的访存特性, 具有较优的整体性能.

## 2 面向 CMP 的可变相联度混合 Cache 结构

### 2.1 CMP-VH 结构

图 1(a) 所示是以 CMP-VH 作为 L2 Cache 的 CMP 平台. 其中,  $\text{Tag}_0 \sim \text{Tag}_N$  为私有 Tag 区, 核独占;  $\text{pData}_0 \sim \text{pData}_N$  为私有数据区, 核独占;  $\text{sData}$  为共享数据区, 所有核共享. 私有 Tag 区基于 LRU 替换策略进行管理, 采用多路组相联结构. 数据区基于 E-Reuse 替换策略进行管理, 使用索引的方式访问. 每个数据区中由 Data 阵列和重用计数表 (Reuse Count Table, 简称为 RCT) 两部分组成, 分别用于存储数据、记录数据项的重用信息及 Tag 项地址. Tag 项和 RCT 项的实现如图 1(b) 所示.

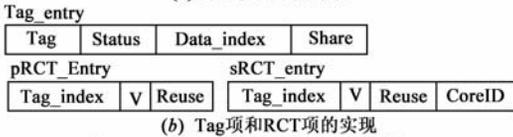
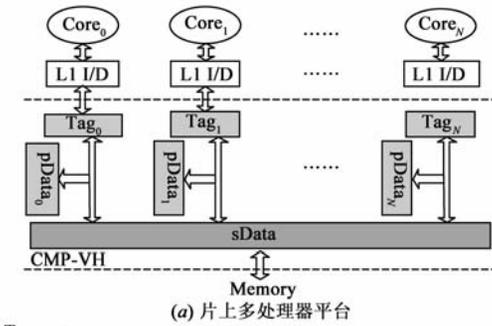


图1 CMP-VH的结构框图及实现

以核  $i$  为例阐述存储访问操作流程.如图 2 所示,首先根据核  $i$  的存储访问地址  $Addr$  访问  $Tag_i$ ,进行 Tag 比较,判断是否命中.如果命中,则读出  $Share$  和  $Data\_index$  指向的数据.如果失效,则启动替换机制选取替换的 Tag 项和数据项,分别称为  $tag\_rp$  和  $data\_rp$ ,用于放置新加载项. $tag\_rp$  按照 LRU 替换原则从  $Tag_i$  中选取,然后根据  $tag\_rp$  是否空闲选取  $data\_rp$ .如果  $tag\_rp$  非空闲,则  $tag\_rp$  中  $data\_index$  指向的数据项即为  $data\_rp$ .否则,表示  $tag\_rp$  当前为空闲 Tag 项,由 E-Reuse 替换策略决定  $data\_rp$  的选取.首先确定私有或者共享数据区,然后顺序搜索选定数据区的 RCT,找到 Reuse 为 0 的项作为  $data\_rp$ .

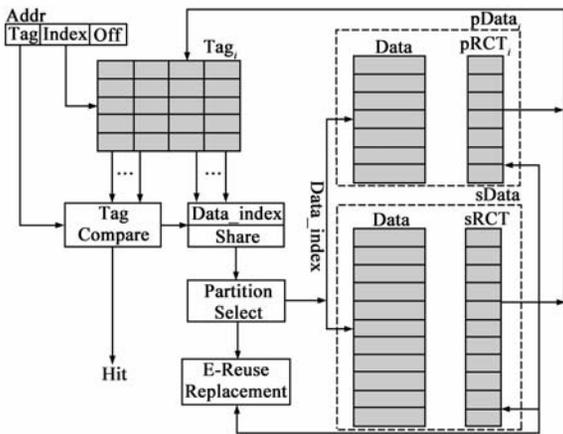


图2 CMP-VH的存储访问流程

### 2.2 E-Reuse 替换策略

CMP-VH 扩展 Reuse 替换策略,通过控制私有/共享数据区的替换频率,实现核间容量划分.根据实现方式的不同,提供隐式和显式两种划分机制.

隐式划分设置私有数据区与共享数据区的替换比例(Private-to-Shared Ratio, 简称为 PSR).假设 PSR 为  $N/M$ ,则 E-Reuse 替换控制进行  $N$  次私有数据区替换后进行  $M$  次共享数据区替换,依此类推. PSR 值越小,对共

享数据区资源的抢占能力就越强,反之越弱.

显式划分限制每个核可以使用的数据项.当核实际使用数据项数小于最大数据项数时,在共享数据区进行替换;否则,在私有数据区进行替换.

现有的静、动态划分机制都可以应用到 CMP-VH 中.本文没有深入探讨共享数据区的划分策略,假设 CMP-VH 仅使用简单地静态隐式划分机制,即所有核等同对待,PSR 值相同且为固定值.

## 3 实验设置

实验基于 Virtutech Simics<sup>[2]</sup>的全系统多核模拟器,参数设置如表 1 所示.CMP-VH 的 PSR 缺省为 1/1,每个核最多可以使用的数据项数等于它的私有 Tag 项数.L2 Cache 的访问延迟使用 CACTI<sup>[3]</sup>进行估算.此外,L2 Cache 远程命中时将不进行存储器访问,而是复制数据.

表 1 模拟参数设置

参数	指标
System	1 Chip, 8 cores, Sparc-V9, Solaris 10
Core Configuration	Single-issue, in-order, no-branch-predictor
L1 I/D Cache	32KB, 2 way, LRU, 64B line, 1 cycle
Shared L2 Cache	1MB, 16way, LRU, 64B line, 20 cycles
Private L2 Cache	128KB per core, 2 way, LRU, 64B line, 10/24 cycles
CMP-VH	Private Tag 4 way, 4K entries per core Private Data 1K entries per core Share Data 8K entries Latency 13/30 cycles
CC Protocol	Snoop based MESI protocol
Main Memory	200 cycles

表 2 测试程序的描述及输入

名称	功能描述	数据输入	分类
fft	快速傅立叶变换	1M 复数	低
lu-contiguous	稠密矩阵分解(优化数据存储),简称 lu	1024 * 1024	中
lu-noncontiguous	稠密矩阵分解(数据块不连续存放,非优化),简称 nlu	1024 * 1024	中
radix	并行 radix 排序,简称 rax	10M 整数	中
cholesky	稀疏矩阵分解,简称 cho	tk15.o	高
ocean-contiguous	海洋模拟(优化数据存储),简称 oc	514 * 514	高
ocean-noncontiguous	海洋模拟(数据块不连续存放,非优化),简称 noc	514 * 514	高
barnes	N-body 模拟,简称 bar	16K particles	高
raytrace	三维场景构建,简称 ray	Ball	高
radiosity	场景中光效果计算,简称 rad	Room	低
water-nsquared	水分子作用力建模,不带空间数据结构,简称 wan	512	中
water-spatial	水分子作用力建模,带空间数据结构,简称 was	512	低
fmv	多层快速多极子算法	1024	低
volrend	Volume renderer,简称 vol	Head-scaled wn2	低

使用 SPLASH-2<sup>[4]</sup>作为测试程序,其功能描述、数据输入如表 2 所示.所有应用分 8 个线程运行,每个线程(wan 和 was 除外)执行约 3 亿条指令,前 1 亿条指令用于预热,后 2 亿条指令进行数据统计.由于应用长度的限制,wan、was 将执行完所有指令,前 1 亿条指令用于预热,后面的进行数据统计.使用失效率衡量 L2 Cache 性能,使用 IPC 衡量系统整体性能.

## 4 结果分析

### 4.1 CMP-VH 性能分析

与传统共享、私有结构相比,CMP-VH 在容量与延迟之间权衡,具有较优的性能.此外,CMP-VH 可以方便地实现 Set 相联度调整和共享数据区容量划分,实现资源的有效利用.图 3 将 CMP-VH 与传统共享、私有结构、Haakon 等<sup>[5]</sup>提出的混合结构的 L2 Cache 失效率进行比较.与私有结构相比,CMP-VH 的 L2 Cache 失效率普遍降低,平均失效率降低 23.37%.与共享结构相比,共享结构下的平均失效率为 1.62%,CMP-VH 结构下为 1.87%,两者较为接近.与 Haakon 结构相比,CMP-VH 的平均失效率降低 6.97%,主要是由于 E-Reuse 替换策略在全局范围选取替换块,资源利用率优于 LRU 替换策略.

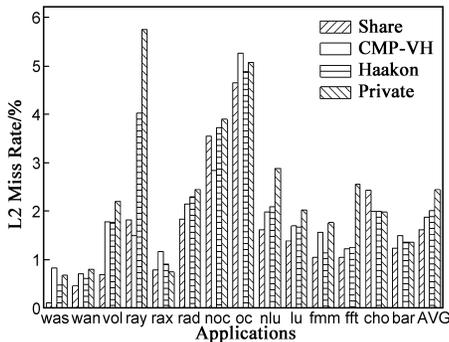


图3 二级Cache失效率的比较

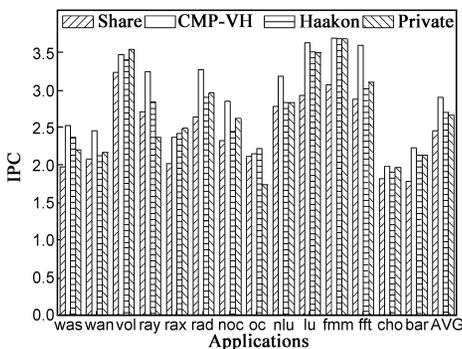


图4 IPC的比较

四种结构下的IPC如图4所示.以共享结构下的平均IPC2.45为基准,私有结构下的平均IPC提高8.64%,Haakon 结构提高 10.20%,CMP-VH 结构提高 18.38%.可见,CMP-VH 能有效提升系统性能.

### 4.2 PSR 的选择

PSR 是静态划分机制的一个重要参数,值越大,在私有数据区进行替换的比例越高,对共享数据区的抢占能力越弱.一般地,给存储访问需求高的应用设置较低的 PSR 值,需求低的应用设置较高的 PSR 值.将 PSR 分别设置成 3/2、1/1、2/3、1/2 和 1/3,我们比较了不同应用下 PSR 对失效率的影响,如图 5 所示.根据共享结构下每千条指令的平均 L2 Cache 失效率的不同,将测试程序分成访存需求高、中、低三类,如表 2 所示.从图 5 可以看出,需求高的 oc、noc 在 PSR 为 1/3 时失效率较低,需求低的 rad、vol 在 PSR 为 3/2 时失效率较低.然而,由于核间存储访问分布存在差异,且应用的访存特性是随时间变化的,PSR 值并不是简单的设置成高或者低就行,如 cho 的 PSR 设置 1/1 时较好.

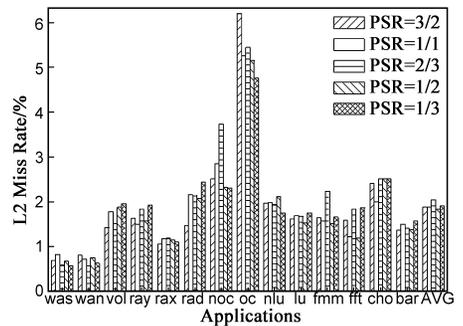


图5 不同应用下,PSR值对CMP-VH失效率的影响

### 4.3 存储开销

与组相联结构相比,CMP-VH 中增加了附加 Tag 资源、双向指针、重用计数器的存储开销.假设地址空间为 32b,块大小为 64B.CMP-VH 的 Tag 项长度为 44b,其中 26b 为 Tag 域,2b 用于 LRU 标识,2b 用于 MESI 协议标识,share 域需要 1b,数据项索引地址需要 13b.私有数据区中数据项长度为 527b,其中 512b 是数据,1b 标识数据项中数据是否有效,12b 存储 Tag 项索引地址,2b 重用计数;共享数据区中数据项还要加上 3b 的核标识.忽略译码器等的开销,CMP-VH 的总存储开销约为  $(44 * 4096 * 8 + 527 * 1024 * 8 + 530 * 8192) \text{b} = 1233 \text{KB}$ ,比共享结构的存储开销(1088KB)增加 145KB,约 13.33%.

## 5 相关研究

私有 Cache 结构的访问延迟低,然而由于每个核可以使用的容量有限,容易造成容量失效,不少学者致力于在私有结构的基础上扩充容量共享.

协同缓冲机制<sup>[6]</sup>、CMP-NuRAPID 结构<sup>[7]</sup>通过控制数据项的迁移和复制实现私有结构上的共享.类似于 CMP-NuRAPID 结构,CMP-VH 也是将 Tag 和数据分离,使用索引的方式访问数据.然而,CMP-NuRAPID 采用距离相关性管理数据,侧重线延迟优化;CMP-VH 基于数

据项的重用频率管理数据,侧重 Cache 资源利用率的提高.随着线延迟的增大,CMP-VH 下一步将考虑非一致性访问的问题.

私有结构上的共享还可以通过预留部分私有空间供其它核使用实现.Haakon 等<sup>[5]</sup>将本地 Cache 分成私有和共享两部分,根据应用需求自适应调整私有、共享部分容量.本文提出的 CMP-VH 结构通过专门的共享数据区实现共享,与 Haakon 等采用 LRU 替换策略管理不同,CMP-VH 使用 E-Reuse 替换策略管理,获得了较好的性能(见 4.1 节).

## 6 结束语

目前 CMP 的最后一级 Cache 结构一般使用 LRU 管理的多路组相联结构,不能很好地适应应用内部、应用之间变化的存储访问特性.本文提出一种基于重用信息管理的可变相联度混合 Cache 结构——CMP-VH. CMP-VH 使用私有的 Tag 阵列,私有与共享并存的数据阵列,结合了私有与共享结构的优势;使用 E-Reuse 替换机制,能够根据应用需求调整组相联度、进行核间容量划分,具备应用自适应能力.SPLASH-2 程序集的模拟结果表明,CMP-VH 的平均失效率与共享结构接近,比私有结构降低 23.37%.同时,CMP-VH 能有效提高系统的整体性能,平均 IPC 比共享结构提高 18.38%,比私有结构提高 8.97%.

下一步研究工作包括:(1)深入研究静、动态划分机制,增强应用的自适应能力;(2)考虑多道程序负载下的优化;(3)探讨非一致性访问对 CMP-VH 结构的影响及性能优化策略.

### 参考文献

- [1] Moinuddin K Qureshi, David Thompson, Yale N Patt. The v-way cache: demand based associativity via global replacement [A]. Proceedings of the 32nd Annual International Symposium on Computer Architecture [C]. Washington, USA: IEEE Computer Society Press, 2005. 544 – 555.
- [2] P S Magnusson, M Christensson, J Eskilson, et al. Simics: a full

system simulation platform [J]. Computer, 2002, 35 (2): 50 – 58.

- [3] Shivakumar P, Jouppi N. CACTI 3.0: an integrated cache timing, power, and area model [R]. CA, USA: Western Research Laboratory, 2001.
- [4] Steven Cameron Woo, Moriyoshi Ohara, Evan Torrie, Jaswinder Pal Singh, Anoop Gupta. The SPLASH-2 programs: characterization and methodological considerations [A]. Proceedings of the 22nd Annual International Symposium on Computer Architecture [C]. New York, USA: ACM Press, 1995. 24 – 36.
- [5] H Dybdahl, P Stenstrom. An adaptive shared/private NUCA cache partitioning scheme for chip multiprocessors [A]. Proceedings of the 13th International Symposium on High-Performance Computer Architecture [C]. Washington, USA: IEEE Computer Society Press, 2007. 2 – 12.
- [6] Jichuan Chang, Gurindar S Sohi. Cooperative caching for chip multiprocessors [A]. Proceedings of the 33rd Annual International Symposium on Computer Architecture [C]. Washington, USA: IEEE Computer Society Press, 2006. 264 – 276.
- [7] Zeshan Chishti, Michael D Powell, T N Vijaykumar. Optimizing replication, communication, and capacity allocation in CMPs [A]. Proceedings of the 32nd Annual International Symposium on Computer Architecture [C]. Washington, USA: IEEE Computer Society Press, 2005. 357 – 368.

### 作者简介



晏沛湘 女,1981年3月出生于湖南新化,国防科学技术大学计算机学院博士生,主要研究方向为微处理器设计及微电子.  
E-mail: peipeixj@nudt.edu.cn

杨先炬 男,1980年8月出生于湖南汨罗,国防科学技术大学计算机学院博士生,主要研究方向为微处理器设计及微电子.

张民选 男,1954年2月生,教授,博士生导师.主要研究方向为高性能计算机系统结构、微处理器设计、低功耗设计及 ASIC 技术等.