

适用于多核环境的混合 Cache 一致性协议

李静梅, 杨鹏飞, 张健沛, 门朝光, 吴艳霞

(哈尔滨工程大学计算机科学与技术学院, 哈尔滨 150001)

摘要: 提出一种适用于多核环境的混合 Cache 一致性协议。该协议采用混合值传播策略, 引入小容量目录 D-Cache, 克服传统监听一致性协议发送数据请求时盲目广播的缺点, 通过数据块状态的扩展, 有效避免乒乓现象的发生。仿真实验结果表明, 该协议能减少测试程序的运行时间, 降低多核处理器私有 L1 Cache 的失效率, 提高系统性能。

关键词: Cache 一致性协议; 多核环境; 目录; 数据块状态; Cache 失效率

Hybrid Cache Coherence Protocol Suitable for Multi-core Environment

LI Jing-mei, YANG Peng-fei, ZHANG Jian-pei, MEN Chao-guang, WU Yan-xia

(College of Computer Science and Technology, Harbin Engineering University, Harbin 150001, China)

【Abstract】 This paper presents a hybrid Cache coherence protocol suitable for multi-core environment. It uses the hybrid value propagation police, introduces a small capacity directory, D-Cache. The protocol overcomes the shortcoming of traditional coherence protocol that data request is undifferentiated broadcasted. It extends data block state so that eliminates ping-pang phenomenon. Simulation experimental results show that the protocol reduces test workload runtime and L1 Cache miss rate, improves the system performance.

【Key words】 Cache coherence protocol; multi-core environment; directory; data block state; Cache miss rate

DOI: 10.3969/j.issn.1000-3428.2011.24.095

1 概述

随着多核技术的深入应用, Cache 一致性问题作为多核处理器设计的关键越来越受到关注, 同时, Cache 一致性协议的研究也不断深入。作为制约多核处理器系统运行效率的关键, Cache 一致性协议的好坏在很大程度上影响着处理器的性能。

传统的 Cache 一致性协议分为基于总线监听的一致性协议和基于目录的一致性协议^[1], 虽然两者各有优点, 但限于设计理念, 它们也存在缺陷。本文在研究 2 类协议的基础上, 提出一种混合的 Cache 一致性协议(MEDSIF), 并在体系结构中设置一个具有小容量目录功能的 D-Cache, 使该协议在继承总线监听协议优点的同时, 选择性地结合了目录协议的优点。

2 经典 Cache 一致性协议研究

2.1 监听一致性协议

监听一致性协议一般用在总线或环形互联结构的多核系统中, 通过广播机制实现一致性事务。使用共享总线互联多个处理器内核的私有 Cache, 内核时刻监听总线是否有请求送达, 所有处理器内核发出的请求以广播方式发送到其他处理器内核节点, 共享总线保证所有处理器内核的数据请求串行执行^[2]。典型的监听协议有 MESI 协议和 MESIF 协议。

监听协议的优点是结构比较简单, 但因为总线是独占性资源, 当总线上互联的处理器内核节点较多时, 协议的效率会急剧下降^[3]。处理器内核的一致性数据请求事务都要无差别地在总线上广播, 并且所有监听到请求的 Cache 控制器都要查找 Cache 中是否包含了请求数据的副本, 即使不包含远程请求数据副本的 Cache 也需要执行查找操作, 无谓地消耗了系统的资源; 同时, 数据请求发出后, 可能有多个私有

Cache 含有请求的副本, 还需要进行仲裁, 造成了额外的仲裁开销; 另外, 一致性协议对于值传播策略的选择单一, 即写无效或者写更新, 并且绝大多数的监听一致性协议都采用了写无效的值传播策略, 这将会导致如果一个数据块副本在 2 个 Cache 中交替更新时, 每次都会使一个 Cache 中副本的状态无效, 造成数据块在 2 个内核之间形成乒乓现象, 引起系统性能的下降。

2.2 目录一致性协议

目录协议使用目录存储 Cache 数据副本的相关信息, 请求的数据在查找目录后以点对点通信方式获取^[4]。目录保证了所有处理器内核对同一数据的访问请求串行执行。目录方式克服了监听协议中无差别发送广播的缺点, 使得不再使用广播机制就可将 Cache 一致性协议扩展到大规模处理器系统中。目录协议以全映射目录、有限目录和链式目录为代表。

目录协议中采取了“以空间换取时间”的策略^[5], 由于需要在高速缓冲中为处理器内核中的 Cache 数据副本创建较多的目录项, 占用了大量的存储空间, 因此在目录较大时, 查询时延也较大。

3 小容量目录 D-Cache

本文协议为克服监听协议无差别发送广播的缺点, 在系统结构中引入了一个目录 Cache——D-Cache, D-Cache 通过互联结构和处理器内核的私有 L1 Cache 互联。带有 D-Cache 的系统结构模型如图 1 所示。

基金项目: 国家自然科学基金资助项目(60873037, 60873138)

作者简介: 李静梅(1964—), 女, 教授, 主研方向: 计算机系统结构, 多核处理器性能优化; 杨鹏飞, 硕士研究生; 张健沛、门朝光, 教授; 吴艳霞, 讲师

收稿日期: 2011-06-28 **E-mail:** lijingmei@hrbeu.edu.cn

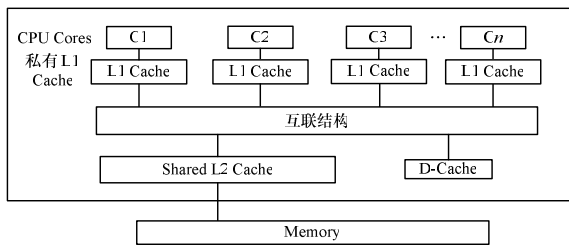


图 1 带有 D-Cache 的系统结构

D-Cache 作为一个小容量目录, 存储的主要内容为: 数据块副本地址及其在所在 L1 Cache 的内核号。任一个数据请求的发送将不再像以前一样, 通过广播发送给所有内核, 而是首先在 D-Cache 中查找数据副本所处的内核号, 之后通过点对点通信发送数据请求。

D-Cache 目录项格式如下:

Address	Condition	Core-number
---------	-----------	-------------

其中, Address 为数据块副本地址; Condition 为数据块副本状态; Core-number 为数据块副本所在内核编号。

4 MEDSIF 协议设计

本文提出的 MEDSIF 协议在传统协议数据块副本状态的基础上进行了扩展, 将状态扩展为 7 个; 同时将传统协议中单一的值传播策略进行改进, 即结合写无效和写更新的混合式策略。

4.1 状态集合

在本文的协议中, Cache 数据块副本可能出现的状态如表 1 所示。

表 1 MEDSIF 协议状态集合

状态	描述
E	Exclusive, 表示仅在本地 Cache 中拥有该块的副本, 且没有被修改过
M	Modified, 表示此时只在本地 Cache 中包含了有效副本, 且此副本已经被修改
DP	Double Primary, 表示在本地 Cache 中拥有该块的副本, 且该块存在另外 1 个副本, 此副本是两者的所有者
DS	Double Slave, 表示在本地 Cache 中拥有该块的副本, 且该块存在另外一个副本, 此副本是两者的从属者
S	Shared, 表示该块未被修改且有 3 个或 3 个以上处理器内核 Cache 包含了该块的副本
F	Forwarding, 表示该块未被修改且有 3 个或 3 个以上处理器内核 Cache 包含了该块的副本, 且远程处理器内核针对该块发出的请求, 只由此状态的处理器内核回应
I	Invalid, 表示该块副本处于无效状态, 如果此时处理器内核需要读取此块数据, 则必须从下级存储器或者其他含有有效数据的 Cache 中读取此数据块

4.2 目录协作

D-Cache 目录中存放除了状态为 S 和 I 的所有数据块副本的目录信息。目录为一个数据块的 DP 和 DS 状态副本分别创建目录项, 状态为 E、M 和 F 的数据块副本对应一个目录项。为一个数据块的 DP 和 DS 状态副本创建目录项, 为消除乒乓现象提供了定位依据。

加入 D-Cache 后, 所有的数据请求首先通过目录进行集中查询, 即以 D-Cache 目录为中心, 源结点请求→D-Cache 查找转发→目的结点返回的方式实现一致性事务。内核发出的一个数据请求被分解为 2 个: 目录请求和数据请求。一致性事务处理流程如下:

- (1)源内核向 D-Cache 发送目录请求。
- (2)如 D-Cache 请求队列未满, 发送请求确认; 如请求队列已满, 不向源内核发送请求确认, 源内核在没有收到确认时, 隔时重发目录请求。
- (3)D-Cache 进行目录查询, 然后发送定点(明确指出目的

结点位置, 非广播)数据请求。如命中目录, 向目的内核发送内核 Cache 数据请求报文; 如未命中目录, 向共享存储器发送共享存储器数据请求。

(4)目的结点接收数据请求, 并向源结点发回响应数据。

4.3 读操作

当处理器内核读本地私有 Cache 命中时, 不修改副本的状态。当处理器内核读本地私有 Cache 未命中或副本状态为 I 时, 向 D-Cache 发送目录查询请求, D-Cache 查询目录, 如果查找目录未命中, D-Cache 向共享存储器发送数据请求, 之后获得数据, 副本状态为 E, 并添加目录项; 如果查找目录命中, 则 D-Cache 根据查询结果定点地向含有副本的目的内核发送包含源请求内核号的数据请求。

其他处理器内核在收到请求后按以下步骤处理:

- (1)查看请求是否是发送给自己的, 如不是则简单丢弃; 如是, 进行下一步处理。
- (2)含有请求数据的目的内核将数据副本以点对点方式发送给源请求内核, 并进行数据副本状态维护。
- (3)目的内核向 D-Cache 发送确认, D-Cache 维护目录。
- (4)源请求内核接收提取数据, 并进行数据副本状态维护。

4.4 写操作

当处理器内核写本地私有 Cache 命中, 且命中除状态为 DP 和 DS 以外的副本时, 不需要发送数据请求, 只在写数据的同时进行必要的副本状态和目录维护, 其规则为:

- (1)如命中 E 状态的数据副本, 状态修改为 M, 并维护目录。
- (2)如命中 M 状态的数据副本, 不需修改状态和维护目录。
- (3)如命中 F 或 S 状态的数据副本, 状态修改为 M, 广播发送写无效请求并维护目录。

当命中状态为 DP 和 DS 的副本时, 根据本协议消除乒乓现象的设计思想, 通过 D-Cache 找到另一个远程处理器内核的副本并将其更新。命中本地 DP 或 DS 状态副本的写操作不需要修改副本状态和维护目录, 但要更新另一个副本的值。

当处理器内核写本地私有 Cache 未命中或副本状态为 I 时, 处理规则与读操作类似, 首先向 D-Cache 发送目录查询请求, D-Cache 查询目录, 如果查找目录未命中, 则 Cache 向共享存储器发送数据请求, 之后获得数据并写值, 副本状态为 M, 并添加目录项; 如果查找目录命中, 则 D-Cache 根据查询结果定点的向含有副本的目的内核发送包含源请求内核号的数据请求。

处理步骤如下:

- (1)其他处理器内核在收到请求后查看请求是否是发送给自己的, 如不是则简单丢弃, 如是, 进行下一步处理。
- (2)含有副本的目的内核将数据副本以点对点方式发送给源请求内核, 响应中包含自己的内核号, 并进行副本状态维护。
- (3)源内核获取数据副本, 写数据及维护状态, 发出值传播请求。根据提供响应内核中此副本状态的不同, 处理过程不同: 如提供响应内核的此副本状态为 E 或 M, 发送写更新请求, 更新另一副本(定点发送, 因为响应中含有提供副本的内核号); 如提供响应内核的此副本状态为 E、M 之外的状态, 只简单发送写无效请求。

- (4)被更新或被无效的内存维护副本值和状态。
- (5)源内核向 D-Cache 发送确认, D-Cache 维护目录。

4.5 状态转换过程

MEDSIF 协议状态转换过程如图 2 所示, 其中, 实线表示处理器内核读写; 虚线表示副本置换。

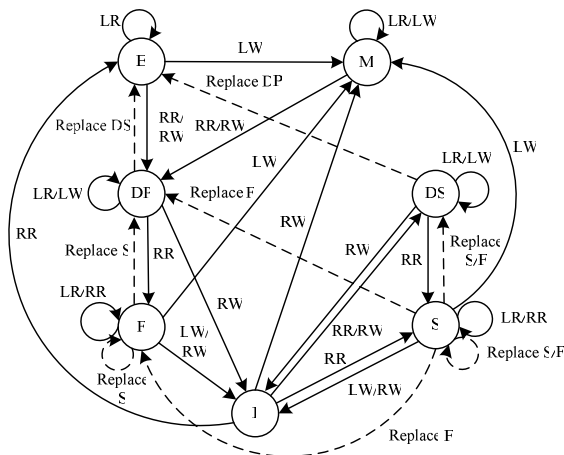


图 2 状态转换过程

状态转换过程中所涉及的操作有: LR: 本地读, 副本在本地内核 Cache 中; LW: 本地写, 副本在本地内核 Cache 中; RR: 远程读, 副本不在本地内核 Cache 中; RW: 远程写, 副本不在本地内核 Cache 中; Replace: Cache 副本替换。

5 仿真实验与分析

为测试新协议的性能, 本文选取 GEMS^[6]模拟器仿真 MESI 协议和 MEDSIF 协议, 进行性能对比。GEMS 是一款多核全系统模拟器, 其上可以运行并行多核程序和操作系统, 能够满足研究人员对计算机体系结构各方面需求的模拟。在 MESI 协议的仿真过程中, 系统中不设置 D-Cache。本文在测试系统中选用交叉开关作为互联结构, 测试系统的各项参数如表 2 所示。

表 2 系统参数

参数	参数值
内核数	8
L1 指令 Cache	64 KB 容量, 2 cycle 访问延迟
L1 数据 Cache	64 KB 容量, 2 cycle 访问延迟
L1 Cache 相连度	4 路组相连
L2 Cache	16 MB 容量, 6 cycle 访问延迟
L2 Cache 相连度	16 路组相连
Cache 块大小	64 Byte

为对两者性能进行精确测试, 采用 SPLASH-2^[7]测试集中测试程序 LU、Ocean、Radix、FFT 和 Water-sp 来测试 2 种协议的性能。测试集的各项参数如表 3 所示。

表 3 测试集参数

测试集	参数值
LU	512×512 matrix
Ocean	258×258 ocean
Radix	1M keys, 基数为 1 024
FFT	256 KB complex doubles
Water-sp	粒度为 512

图 3 给出了在 2 种协议环境下运行 5 个测试程序的运行时间对比。测试结果以 MESI 协议环境下测试程序的运行时间为基准。以平均值看, MEDSIF 协议环境下测试程序平均运行时间比 MESI 协议少 3.92%。图 4 给出了运行测试程序的 L1 Cache 失效率对比, 测试结果同样以 MESI 协议环境下的 L1 Cache 失效率为基准。以平均值看, MEDSIF 协议环境

下 L1 Cache 失效率比 MESI 协议低 2.88%。

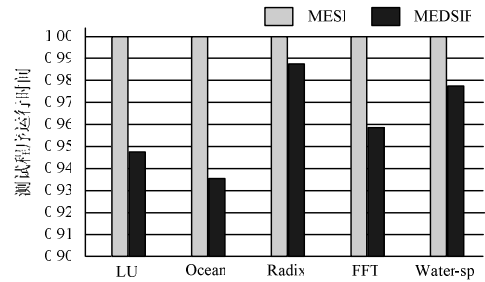


图 3 程序运行时间对比

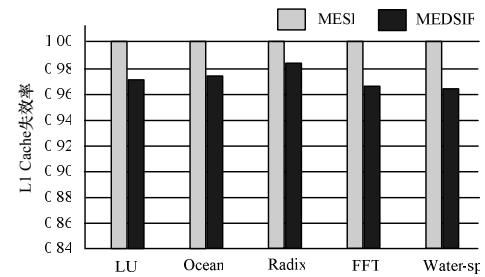


图 4 L1 Cache 失效率对比

MEDSIF 协议与 MESI 协议相比, 运行测试程序所需的时间更少, 具有更高的系统运行效率; 由于改进了乒乓现象, 因此减小了 L1 Cache 失效率, 从总体上提高了系统的性能。

6 结束语

随着多核处理器应用的不断深入, 如何提高多核处理器性能成为重要的研究课题。根据处理器体系结构对一致性协议的改进可以有效提高系统的性能。本文提出的 MEDSIF 协议结合目录协议和监听协议的优点, 克服了监听协议无差别广播一致性事务的缺点, 并通过扩展已有监听协议状态集合, 有效地避免了 Cache 乒乓现象的发生, 提高了系统的性能。但是, 本文在协议设计和模拟验证阶段没有考虑能耗和安全等问题, 这些问题有待今后深入研究。

参考文献

- [1] 薛燕, 樊晓妮, 李瑛. 多处理机系统中数据 Cache 的一种优化设计[J]. 微电子学与计算机, 2004, 21(12): 191-194.
- [2] Meng Jiayuan, Skadron K. Avoiding Cache Thrashing due to Private Data Placement in Last-level Cache for Manycore Scaling[C]//Proc. of IEEE International Conference on Computer Design. Lake Tahoe, California, USA: [s. n.], 2009.
- [3] 黄安名, 高军, 张民选. 多核处理器 Cache 一致性协议关键技术研究[J]. 计算机工程与科学, 2009, 31(z1): 104-107.
- [4] 王铮, 张毅. 改进的基于目录的 Cache 一致性协议[J]. 计算机工程, 2009, 35(1): 87-89.
- [5] Zhao Hongzhou, Shriraman A, Dwarkadas S. SPACE: Sharing Pattern-based Directory Coherence for Multicore Scalability[C]//Proc. of the 19th International Conference on Parallel Architectures and Compilation Techniques. Vienna, Austria: [s. n.], 2010.
- [6] Marty M R. Cache Coherence Techniques for Multicore Processors[D]. Madison, USA: University of Wisconsin-Madison, 2008.
- [7] Woo S C, Ohara M, Torrie E, et al. The SPLASH-2 Programs: Characterization and Methodological Considerations[C]//Proc. of the 22nd Annual International Symposium on Computer Architecture. Santa Margherita Ligure, Italy: [s. n.], 1995.