

处理器片上渗透缓存蕴含的时间与空间及时局部性

胡九川¹,程建聪²,万良易³,吴楠士¹,叶笑春⁴,严 龙⁴

(1. 北京交通大学计算机科学与技术学院,北京 100044;2. 教育部教育管理信息中心,北京 100816;3. 中国电子科技集团公司第 32 研究所,上海 201800;4. 中国科学院计算技术研究所计算机体系结构国家重点实验室,北京 100190)

摘 要: 处理器片上寄存器的分布形态与数量规模对处理器的整体计算性能有直接影响,这种影响表面上看是波及处理器片上缓存结构的改进和优化,本质上是时间要素与空间要素交织在一起的综合反映. 因此,从时间和空间上确保处理器内核对片上缓存的局部化访问必将进一步提高处理器的整体计算性能. 为了认识处理器片上缓存中存在的时间与空间及时局部性,以由传统缓存耦合而成的渗透缓存为工具来分析处理器内核访问片上缓存的时间与空间局部性,仿真实验表明渗透缓存因具备容纳时间与空间局部性的结构提高了处理器访问片上缓存的命中率,客观上缩短访存延迟,从而为提高处理器性能创造了有利条件.

关键词: 渗透缓存;时间局部性;空间局部性;片上数据流转

基金项目: 国家自然科学基金(No.62202451)

中图分类号: TP302.1

文献标识码: A

文章编号: 0372-2112(2024)10-3589-11

电子学报 URL: <http://www.ejournal.org.cn>

DOI: 10.12263/DZXB.20231115

The Time and Spatial Just-in-Time Locality of on-Chip Percolation Cache

HU Jiu-chuan¹, CHENG Jian-cong², WAN Liang-yi³, WU Nan-shi¹, YE Xiao-chun⁴, YAN Long⁴

(1. School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China;

2. Center for Information of Education Management, Ministry of Education, Beijing 100816, China;

3. The 32nd Research Institute of China Electronics Technology Group Corporation, Shanghai 201800, China;

4. State Key Laboratory of Computer Architecture, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China)

Abstract: It is believed that the arrangements and amounts of the registers in processor chip have much heavy impact on the operation speed of the processor, which has induced the improvement of the structure of the on-chip cache, whose central task is to realize the fast access to the data in registers in the term of time and space. This kind of fast access to the register can be investigated via the access process in which the data and structures in on-chip cache are accessed. By introducing the a new on-chip cache percolation cache, we prove that the existent of the time and spatial just-in-time locality which the percolation cache equips has contributed much to shorten the memory access delay by raising the hit rates when processor core accesses the percolation cache.

Key words: percolation cache; temporal locality; spatial locality; data flow on-chip

Foundation Item(s): National Natural Science Foundation of China (No.62202451)

1 引言

为了提高处理器的性能,处理器体系结构内的缓存体系需要不断地改进,以适应处理器高速运行和确保计算过程安全的现实需要. 这些缓存体系包括处理器片上的寄存器和缓存. 其中,片上缓存面临持续稳定地向处理器流水线提供指令和数据的需求. 为了达到这样的目的,人们提出了让存储在计算机内的指令和

数据流动起来的构想^[1-4].

指令和数据在处理器内部的流动的构想首先萌发于处理器的流水线设计^[5,6],流水线上指令和数据的流动是遵循一定的次序在时钟节拍的控制下展开的. 但是,随着进入处理器的指令和数据的数量大幅增长,单位时间内处理器处理指令和数据的数量便日益成为衡量处理器吞吐能力的外在指标,这一指标表明时间要

素已成为评估处理器运行性能的一个鲜明的视角。为了进一步提高处理器性能,提出了指令在处理器内部乱序执行的设计思想,进而将单纯的流水线设计予以优化,即在指令解码完成以后,流水线中指令的执行顺序被调整,让满足执行条件的指令优先执行,然后在流水线的末端将指令执行的顺序予以恢复。这样处理器内部以寄存器为代表的缓存的数量规模逐渐扩大,以致存储空间占据了处理器芯片的大部分面积,故而空间要素也逐渐地成为衡量处理器设计质量的外在指标。抛开具体的设计细节,从思想方法层面就时间和空间要素对处理器的设计进行深入研究已经成为必然的趋势。

处理器流水线中的乱序执行设计思想直观上看起来给人以“乱”的表象,然而在“乱”的表象后面隐藏的是遵循由指令传递出来的程序执行的内在逻辑,是“乱”中有序,是处理器流水线“响应”计算机程序内在逻辑的表现。这样能动的处理器性能改进反映了一种新的认识:提高处理器自身运行性能最终要使处理器的执行顺序必然要同计算机程序的内在执行逻辑顺序合拍或融合。

然而,在功能和结构与处理器内核紧紧靠在一起的处理器的片上缓存的设计模式目前没有“响应”这样的合拍要求,片上缓存的设计仍然停留在静止的被动状态^[5,6],即处理器片上缓存仅仅是一个指令和数据进入处理器内核的中间驿站:从处理器内核出来的指令和数据停留在缓存里只是被动等待再次进入处理器内核或计算机内存的时机;从计算机内存出来停留在缓存里的指令和数据同样地是被动地等待进入处理器内核的时机,这些被动的等待具有很大的盲目性和随机性。

因此,应该研究消除指令和数据进出处理器片上缓存的盲目性和随机性的方法。文献[1]给出了一个让指令和数据在处理器片上流动起来,减少指令和数据驻留片上缓存的盲目性和随机性的、主动满足处理器内核访存需求的渗透缓存模型。仿真实验表明,让指令和数据在处理器片上缓存中流动起来提高了处理器内核的访存命中率,在适应处理器内核访存需求上取得了进展。

当指令和数据在处理器内核片上缓存中随处理器内核的访问流动起来后,时间和空间要素的重要性日益显著。正如流水线乱序执行设计思想所表明的那样,按照指令所体现的程序中的内在逻辑,处理器内核所访问的指令和指令之间、数据和数据之间、指令和数据之间存在一定程度上的局部关联,访问指令和数据的先后顺序突显的是时间要素,到处理器片上缓存里哪些地方访问哪些指令和数据突显的是空间要素。为了使处理器内核所访问的指令和数据在进出处理器内核

的时候顺畅有效地流动起来,必须进一步研究分析渗透缓存的结构和功能在将时间要素和空间要素综合起来的过程中所发挥的作用。

2 时间与空间及时局部性

计算机存储程序原理与指令集所表达的计算逻辑密切交织在一起,以指令集为灵魂的计算机处理器结构展现出这交织的实际效果,特别是在处理器片上缓存上,这种交织具有明显的时间与空间特征。采取存储程序方式,包含计算逻辑的计算机程序在运行之际先期进驻处理器片上缓存,使处理器内核可以“近距离”地访问程序中的指令和数据,从而缩短了处理器内核获取数据的空间距离,减少了处理器内核获取数据的时间消耗。

但是,处理器内核对片上缓存内的指令和数据访问是随处理器内核的运行展开程序内包含的计算逻辑而发生的,先期进驻片上缓存内的指令和数据不一定是处理器内核按照计算逻辑所需要访问的指令和数据。造成这样的处理器内核访存“错位”的根源是安置程序指令和数据于存储空间内的方式与方法,指令和数据在片上缓存内的分布正是这种“错位”的集中反映。

事实上,甚至在计算机程序产生以前,计算逻辑便以特定的符号或编码形式首先出现在人脑中,为了将这些符号或编码保存在计算机内以实施计算,这种符号或编码形式最终由计算机内电磁材料的电磁性态(如高低电压、磁针指向等)来具体呈现。容易使人忽视的是这种呈现是受人的控制的。这种受控的存储方式,尽管实现了从人脑到电磁材料性态的转移,但是无法改变对数据信息内涵的解释,仍然需要人来承担这样的客观现实。所以包含在程序内的计算逻辑更无法用这种受控的存储方式呈现出来。

从这个意义上讲,计算机的计算并不能涵盖数据信息加工处理的全部环节,特别是那些需要人参与对数据信息进行解释的环节。从结绳记事开始,人类都是通过特定物质材料的性态来表述和传递符号或编码,这些符号和编码如何指向数据信息则完全是由人来决定。因此,存储程序方式并不完整代表计算机的工作机制而是表现出机制的一个侧面。在根本上,计算是确定客观事物相互联系的一类特殊的人类实践活动,而确定事物相互联系并非离开计算或计算机就不能实现。在以计算机为核心的信息技术领域,计算活动是思维活动与计算机这样的辅助工具融合在一起展开的;这个展开的本质是以数的四则运算和形式逻辑为基本手段的逻辑推演,其目标是确定客观事物的特定联系。

既然是辅助人类实践的计算工具,逻辑推演的效

率,即计算机的计算效率必须在时间和空间上满足人类的要求.在处理器的片上缓存内,处理器内核访问指令和数据的“错位”现象应该尽最大程度地予以控制.在计算机中,人的计算活动是操控电磁材料的性状来展开完成的.在处理器芯片内高低电位信号的转换正是这样的操控;一组相关联的电信号在这样的转换操控中落实处理器指令的计算职能,实现指令的计算意图.在这样的性状改变过程中,时间和空间进入人们的视野,成为认识性状特点及其变化过程的自然视角.单位时间内电磁性状改变的量被抽象成为评价计算性能的一种外在的指标,它掩盖了性状改变过程及其涉及的范围,仅表明性状改变的速度和规模,属于量的范畴内的指标,与计算的内在规定无关.事实上,性状的变化及其改变速度与计算逻辑之间不存在直接联系,即计算机执行计算的速度与计算的正确性和有效性不存在直接的逻辑联系,这与高速运转的传动轴不能决定汽车行驶的正确方向是同样的道理.

计算机处理器的指令集、指令之间的逻辑衔接表达了蕴含在程序内的计算逻辑,包含这些复杂计算逻辑的程序在编译器的协助下构成了与汇编语言、机器语言之间的联系,在硬件层面转化为按一定控制规程联系起来的指令序列,最终这些指令序列表现为受操控的电磁材料的一系列电磁性状以及性状的转换;转换弥漫的空间受电磁材料性质的制约,不同的性状序列分布在这制约所决定的空间中.由于指令的执行需要把涉及的性状序列转移到空间中的不同区域,那些控制性状转移的条件或机制自然成为研制处理器芯片的核心内容.

追求性状转移的效率自然使人们把时间和空间联系起来.文献[7]指出,在当前x86处理器中,由于精简指令流水线比复杂指令流水线效率高,有必要将复杂指令译转换成精简指令,为此需构建一个微操作缓存(micro operation cache)暂时存放处理器运行过程中近期(temporal)解码的指令,发挥时间及时局部性的优势潜力;同样地,发挥靠近处理器核的片上缓存的性能优势,要求人们有必要对片上缓存空间实施优化.文献[8]给出了一个归并相似访存行为^[9]的数据预取工具,文献指出,相似的访存行为导致数据在缓存中存储冗余,从而需要优化高性能片上缓存空间的使用,在性能和存储之间达成平衡.自然地,当空间范围超过一定的限度,即涉及转换的性状序列在空间中分布的广度超出了合理的限度、超出了一定的局部性区域,那么计算机执行指令的时间消耗就成为研制处理器芯片必须面对的问题.在多核体系中,人们通常采用就地计算策略(near-date computing paradigm)将数据处理的职责交付给被处理数据所临近的处理器内核,而就地计算策

略的有效性非常依赖数据在空间上的紧密关系,这些关系紧密的数据往往共处一个局部性区域^[10].这个问题的存在反而意味着如果在电磁材料性状转移所涉及的范围没有超出一定的限度,即具有局部性,时间因素是可以忽略的.因此,在处理器芯片中实现有效的局部性区域必然推动计算性能的改善.

处理器内核的结构与布局受指令集规约,必然占据一定的物理空间,空间中充满了由集成电路构成的基本计算装置,这些装置构成处理器内核.在人预先制定的控制规程下,处理器内核中电子材料的电气性状被改变,将每一条指令意图转化为一组电磁性状的改变过程.如此这般,程序中每个指令的计算意图都可以在这个由处理器内核所占据的空间内转换成相应的电磁性状变换序列,计算过程就是这样展开的,建立起的从“虚”的指令计算意图到“实”的电磁材料电磁性状序列之间的联系,借助的是电磁的物理性质,而人在幕后规划这一切.

电磁性状迁移是计算过程的“主旋律”,从处理器内核之外往内核里面迁移电磁性状是处理器运行的自然展开,也是处理器运行的必然条件.任何计算机程序中的指令或数据都必须转换成一组电磁性状变换序列,在处理器内核之外的特定区域“待命”等候进入处理器内核.由于一个程序指令和数据的数量在理论上是无限的,处理器片上容量有限的缓存内根本无法全部接纳它们,只能把表现指令和数据的绝大部分性状维持在硬盘或内存之中,而将一小部分安置在片上缓存这样特定区域.这片上缓存距离处理器内核最近,其上“局促”的存储空间便天然形成了合理的局部性区域,这种局部性使得指令数据进入处理器内核的过程大幅度缩短,时间消耗不能成为影响计算性能的主要因素.

正如前面所述,处理器片上缓存能够营造的计算性能优势会因计算逻辑的内在衔接在片上缓存内的“错位”而丧失.由于计算机程序中的指令和数据在根本上是串行进入处理器内核被解码后送上流水线的,在此过程中很可能出现指令拟处理的其他指令或数据并不是紧随其后而来到片上缓存,可能它们还滞留在内存或硬盘上.此刻必然要到内存或硬盘中去获取它们,将代表它们的电磁性状变换迁移到片上缓存中来,再送进处理器内核中去.在这种情况下,处理器片上缓存天然具备的局部性优势遭到了破坏.

从抽象的角度看,处理器内核的解码器和流水线上“流动”的是与指令和数据相对应的电磁性状序列,这样的处理器内核的内流动态势当然需要在片上缓存中建立起相应的外流动态势,让指令数据在缓存中流动起来,甚至从硬盘到内存再到片上缓存如流水般流

动起来,形成内流动和外流动的合理衔接.只有这样,处理器片上缓存具备的局部性优势才可以得到维持和保护,有待处理器内核访问的指令和数据可以提前迁移到与处理器内核紧邻的片上缓存之中,形成有助于提高处理器内核访问片上缓存效率的及时局部性环境,削减处理器内核跑到相对遥远的内存中寻找数据造成的访问延迟.

于是,处理器片上营造及时局部性环境承担起了改进计算性能的责任,是结合程序内计算逻辑的内在规定挖掘硬件性能优势的新方法,是实现硬件程序化的新手段.

在这种情况下,处理器片上及时局部性就是一种处理器体系结构中有利于处理器内核访问与其紧邻的片上缓存获取指令和数据的一种性能特征.由于内核访问片上缓存内各个指令或数据所需的时间消耗几乎接近,那么访存命中率便成为这种性能特征的最重要的指标.在这种条件下片上缓存紧邻处理器内核,其中的指令或数据与内核的距离没有内存与内核的距离那样遥远,内核可以就近及时地访问指令或数据已经成为现实,于是内核访存片上缓存的命中率就是衡量处理器体系结构效能特性的核心指标.

显然,在及时局部性环境中被迁移的指令或数据满足处理器内核的访存需要具有决定作用.这种决定作用必须在综合如下三个方面因素的前提下予以分析研究:第一,片上缓存的容量与结构;第二,来到片上缓存中指令或数据之间的逻辑关系;第三,指令或数据在片上缓存中的分布.在实践中,片上缓存的容量与结构是设计选择的结果,当然可以在不同的设计选择的条件下,研究片上及时局部性的变化规律,这方面的工作自然属于一个更大的范围.为了顺利地开始这大范围的研究,可先从小的范围着手,即在确定一种设计选择的情况下,考察片上及时局部性与片上数据特征之间的关系.于是来到片上缓存中的指令或数据之间的逻辑关系及其在片上缓存中的分布形态成为研究的主要对象.

来到片上缓存中的指令或数据之间的逻辑关系在根本上是由程序中的计算逻辑所决定.尽管编译器在组织程序代码和操作系统在规划程序运行环境的时候对指令或数据之间的逻辑关系有所调整,但是仍然可以从时间和空间两个最基本的要素来分析处理器内核访问片上缓存的行为特点,透视来到片上缓存里的指令或数据之间的内在联系,以控制来到片上缓存中的指令或数据的分布形态,使得片上及时局部性环境更加有序合理,从而提高处理器内核的访存命中率.

处理器内核访问片上缓存的根本依据只能是隐含在程序代码指令背后的计算逻辑中,这个计算逻辑必须要在一个具体的特定程序运行环境中得到展开以实现计算目标,从而这计算逻辑必然受到具体的计算机

体系结构要素的影响.当程序代码中包含的所有指令和数据被编译器在一个线性地址空间内分配了逻辑地址以后,这些逻辑地址在根本上决定了计算过程中的执行步骤.计算过程中除了按顺序线性执行的步骤以外,必然要增加在不能按顺序线性执行的时候在不同的指令或数据的地址上进行跳转的步骤.如果说按顺序线性执行的步骤意味着处理器内核访存行为具有小范围空间意义上的地址切换色彩,那么在不同指令和数据地址上的跳转切换则意味着处理器内核的访存行为具有大范围空间意义上的地址切换色彩.如果小空间范围或大空间范围意义上的地址切换出现极端特殊的情况,即程序的执行步骤反复切换回一个固定地址或地址段上,甚至连续保持在一个固定地址或地址段上,那么这种条件下的地址切换意味处理器内核的访存行为具有鲜明的时间色彩.因此,那些被迁移到处理器片上缓存内的指令和数据具有时间及时局部性和空间及时局部性,这是客观事实,只是不同条件下这些及时局部性有程度上的差异.

从国内外研究的情况看,对处理器片上缓存的及时局部性内涵的认识,经历了一个从感性的相对模糊到理性的逐步清晰的发展过程.实际上,计算机中央处理器内核中的片上寄存器和计算机内存的出现蕴含了及时局部性产生的萌芽,直到处理器片上缓存的出现,及时局部性的内涵开始出现在人们的研究意识中.硬件转移预测^[5,6]就是利用及时局部性的典型代表.文献[2]给出了一个在片上缓存靠近多个处理器内核条件下创建片上局部性环境的设想.文献[3]给出了一个计算体系构想,让指令和数据在这个体系内的及时局部性环境中逐步迁移到处理器内核的周围.文献[1]介绍了一个让指令和数据在处理器内核周围流动起来的片上缓存结构.所有上述工作都是围绕营造及时局部性环境机制展开的.

由于在指令和数据从内存被迁移到片上缓存的过程中,它们在迁移的意义上并不存在任何差别,只需要为这些指令和数据在片上缓存中安排合适的存储位置即可,所以为了研究的方便,指令和数据可以统称为数据,可以将指令视为特殊的数据.

3 渗透缓存

揭示来到片上缓存的数据之间存在的联系需要为这些数据的到来提供安置它们的存储空间,并且需要让这些数据动起来,在数据运动的过程中观察数据之间的关联关系.在文献[1]中给出了一个让迁移到片上缓存里的数据随处理器内核的访问在该缓存内有序地流动起来的新型片上缓存,称之为渗透缓存(如图1所示).在渗透缓存中,被访问过的数据乃至重复访问的数据被安置在称为泉吸缓存的存储区域,那些地址和已经被访问了的数

据的地址相邻或距离不远的的数据被安置在称为泉涌缓存的存储区域. 当泉吸缓存区域内的数据不再为处理器内核所需要, 则可以将这些数据迁移回内存. 于是, 随着处理器内核展开访问缓存的过程, 迁移来的数据在渗透缓存内形成一个数据流动的漩涡, 进入这个漩涡的数据沿着泉涌缓存区域的各层级向渗透缓存顶端流去, 到顶后折回再从泉吸缓存顶端沿着泉吸缓存区域内的各层级顺势流下来.

由于停留在泉吸缓存区域的数据有被处理器内核再次访问的可能, 所以这些数据具有显著的时间局部性特征; 同理, 由于驻留在泉涌缓存区域数据的地址和已经被处理器内核访问的数据的地址靠近存在被访问的可能性, 所以这些数据具有显著的空间局部性特征.

渗透缓存由两个传统缓存耦合而成, 在原理上该渗透缓存可以具有多个层级, 为了论述的便捷, 本文仅以具有三个层级的渗透缓存为示例(图 1), 介绍渗透缓存蕴含的时间与空间及时局部性.

一般地, 当前处理器内核所访问的数据称为访存焦点数据. 因为围绕在访存焦点数据周围的数据存在被处理器内核及时访问的可能性, 所以这些数据和访存焦点数据构成一个及时局部组, 及时局部组以访存焦点数据为中心形成一个对称结构, 作为一个整体被迁移到渗透缓存之中.

及时局部组蕴含了时间与空间要素. 处理器内核所访问的焦点数据一定被存贮在内存中的某个存储区域内, 该区域具有唯一的地址, 与该地址相邻靠近的地址上的存贮区域所存储的数据存在被处理器内核在不久的将来访问的可能性. 这里数据空间位置的靠近和被处理器内核接连访问的可能性都揭示了一个客观存在: 以访存焦点数据为中心的数据组构成了一个局部性环境. 一方面因存储数据的位置临近而呈现出空间要素, 另一方面因处理器内核对数据访问的接续性而呈现出时间要素. 空间要素和时间要素是在处理器内核的访存过程中逐渐地展现出来, 这样的展现过程自然会通过数据在片上缓存内的流动中集中地表现出来. 渗透缓存的结构安排正是出于让数据流动起来的考虑而产生的.

图 1 是渗透缓存原理和及时局部组的示意图, 其中地址 4 是处理器核的访存焦点, 相邻地址 3 与地址 5、地址 2 与地址 6、地址 1 与地址 7 构成内存中一个对称的及时局部组. 当地址 4 上的数据被迁往渗透缓存中的泉吸缓存的顶端时, 及时局部组内的其他数据被分别迁往渗透缓存中的第一级、第二级、第三级泉涌缓存. 由于地址 4 的及时局部性程度最高, 与之临近的地址 3 和地址 5 的及时局部性程度也比较高, 所以地址 3 和地址 5 上的数据被迁移到泉涌缓存的第一级. 如果处理器内核在随后的访存中, 其访存焦点仍然是地址 4, 那么该

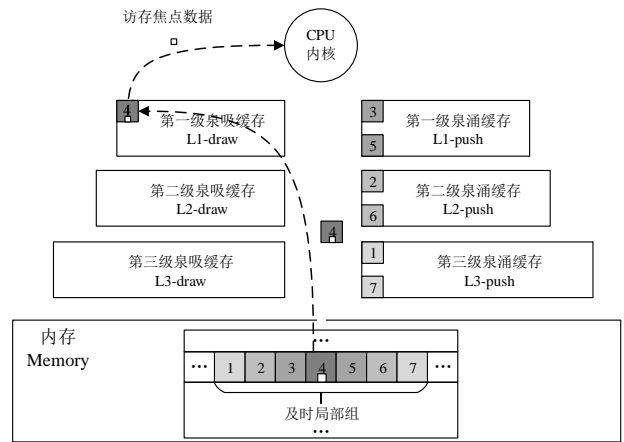


图 1 渗透缓存结构及数据迁入规则示意图

地址的及时局部性仍然保持最高; 如果访存焦点不是地址 4 而是地址 5, 那么它的及时局部性程度变为最高, 地址 4 的及时局部性降低. 此刻, 地址 4 中数据可以停留在原地或被降级迁往渗透缓存中的第二级泉吸缓存, 地址 5 中的数据被升级迁往第一级泉吸缓存, 地址 6 中的数据被升级迁往第一级泉涌缓存, 地址 7 中的数据升级被迁往第二级泉涌缓存, 如此这般, 数据在渗透缓存中开始成渗透式的流动趋势.

数据在渗透缓存中流动是在访存焦点的牵引之下展开的, 在访存焦点的切换中形成多次数据流. 这些数据流有一个显著的区别: 如果处理器内核在渗透缓存中没有命中其要访问的数据, 则处理器内核直接前往内存中获取其要访问的数据. 把在这种情况下发生的一系列数据迁移步骤称为冷渗透, 并把因此形成的数据流称为冷流; 把处理器内核在渗透缓存中命中其所要访问的数据而引发的一系列数据迁移步骤称为热渗透, 并把因此形成的数据流称为热流. 于是数据在渗透缓存中的流动过程由冷渗透和热渗透两种流动方式相互交织构成, 特别是热流中的数据再次被处理器内核访问后, 这些热流数据会在渗透缓存中形成数据流的漩涡, 从而形成漩涡.

从数据在渗透缓存中的流动形态上看, 泉涌缓存中的数据流动更多地呈现出空间及时局部性的特征; 泉吸缓存中的数据流动首先更多地呈现出时间及时局部性的特征. 进一步从数据流自身角度来看, 冷流呈现出相对明显的空间及时局部性的特征, 热流呈现出相对明显的时间局部性的特征. 但是, 因为在泉吸缓存被命中的数据诱导出在泉涌缓存上的新一轮数据迁移热流, 所以数据在渗透缓存中的热渗透实际履行了将时间转换为空间的职能; 同样地, 当泉吸缓存中的数据被处理器内核命中而再次被放置在泉吸缓存的顶部的时候, 这新一轮的数据迁移热流也履行了的将空间转换为时间的职能. 事实上, 冷渗透过程中, 访存焦点数据被置于泉吸缓存顶端之际, 空间向时间转换就已开始.

综合起来,渗透缓存中的冷流和热流交织而形成的片上缓存内的涡流推动了空间向时间和时间向空间转换.认清从片上缓存产生的历史过程后就不难理解时间和空间之间发生的转换.一般地,为了提高处理器的计算性能,人们在处理器内核和内存之间增添片上缓存的做法实际上就是在利用缓存空间将时间问题转化为空间问题.渗透缓存的出现将时间和空间相互转化的内涵展示得更加清晰完整.当前,伴随处理器运行速度提高的是处理器片上的寄存器所占据的空间越来越多,反映了这样的时间换空间和空间换时间是存在的.

4 仿真实验

本文使用 modelsim 10.1a 仿真工具来模拟数据在内存和片上渗透缓存之间的迁移过程,包括在渗透缓存中热渗透和冷渗透的展开过程.图2展示了仿真系统的基本构成,其中处理器模块、缓存模块、内存模块、渗透次数收集模块为核心仿真组件.处理器模块解析程序中指令和数据的地址、生成代表处理器内核访问内存的访存请求;缓存模块处理数据的存储替换、组织渗透并向渗透次数收集模块传递每次渗透结束的信号;内存模块处理数据替换、组织渗透和数据存储;渗透次数收集模块统计单轮渗透次数信号,在收到一轮完整的渗透信号之后,向处理器汇报,使处理器开始下一轮工作.

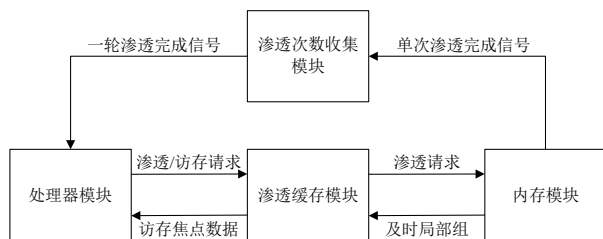


图2 渗透数据迁移仿真平台结构

自然地,渗透缓存中反映出来的时间与空间及时局部性,可能因在处理器上运行的程序不同而不同,但是在处理器上运行同一程序的条件下,渗透缓存上具备的时间与空间及时局部性应该保持相对稳定、呈现一致的变化态势,这种相对稳定性或一致的变化态势不会因为渗透缓存的配置参数的改变而改变.

所以,为了证明上述判断,需要在不同的渗透缓存配置条件下来运行同一个程序.渗透缓存由两个分别称为泉吸和泉涌的传统缓存耦合而成.一般情况下,泉吸和泉涌缓存在各层级的容量是相同的,即泉吸缓存容量和泉涌缓存容量的比为1:1.在仿真实验中,渗透缓存由三层缓存组成,为了验证在运行同一个程序的条件下渗透缓存中的时间与空间及时局部性存在的稳定性,渗透缓存中每层的泉吸缓存和泉涌缓存容量的

配比设置为如下7种情形:1:7、2:6、3:5、4:4、5:3、6:2、7:1,并保持每层泉吸缓存容量和泉涌缓存容量的总容量保持不变;其中渗透缓存的第一级L1由两个总容量为16KB的一般缓存耦合而成,分别是第一级泉吸缓存(L1-draw)和第一级泉涌缓存(L1-push);类似地,渗透缓存的第二级L2由总容量为32KB的一般缓存耦合而成,分别是第二级泉吸缓存(L2-draw)和第二级泉涌缓存(L2-push);渗透缓存的第三级L3分别由总容量为64KB的一般缓存耦合而成,分别是第三级泉吸缓存(L3-draw)和第三级泉涌缓存(L3-push).泉吸缓存由第一级L1-draw、第二级L2-draw、第三级L3-draw组成,泉涌缓存由第一级L1-push、第二级L2-push、第三级L3-push组成,参数设置如表1所示.

表1 渗透缓存基本参数设置

渗透缓存	容量大小/KB	块大小/B	每组块数
L1	32	32	128
L2	64	32	128
L3	128	32	128

处理器内核在访存时,沿着L1-draw、L1-push、L2-draw、L2-push、L3-draw、L3-push的层级顺序进行访问,每次访存均可能存在两种情况,访问命中与未命中,命中后为该层命中次数加1,未命中则沿着访问预先设定好的层级顺序访问下一层级,为下一层级记访问次数加1,因此相对来说,上一层级的访问次数减去命中次数为下一层级的访问次数,每层按照命中次数和访问次数即可计算该层级的命中率.

本文的仿真实验使用3个具有典型代表意义的测试程序 Wordcount、Cholesky、LU 来验证渗透缓存具备的时间与空间及时局部性,实验结果如表2~4所示.

4.1 测试程序 Wordcount

由表2可知,传统缓存结构的整体缓存命中率为88.04%,高于渗透缓存(泉吸缓存的容量和泉涌缓存的容量的比为1:7时)的整体缓存命中率83.88%,且传统缓存的L1层命中率72.23%高于渗透缓存L1-draw与L1-push命中率之和56.64%,但随着渗透缓存结构泉吸缓存与泉涌缓存的容量比逐渐升高,渗透缓存结构的L1和整体缓存的命中率逐渐高于传统缓存结构的L1和整体缓存的命中率,实验结果表明,渗透缓存存在一定优势,即可以通过自由调节泉吸缓存与泉涌缓存的容量比,以提高命中率.另外,表2中的数据表明,当泉吸缓存的容量和泉涌缓存的容量的比为1:7时,访存命中大部分集中在泉吸缓存内,其中泉吸缓存第一级的命中率为48.39%,第二级的命中率为32.55%,第三级的命中率为35.82%;而泉涌缓存的访存命中率比较低.在泉吸缓存容量远远少于泉涌缓存的情况下,命中大部分集中在泉吸缓存中,说明泉吸缓存具备比较突出

表 2 Wordcount 测试集缓存命中情况

缓存结构及容量比	缓存层次	访问次数	命中次数	命中率/%	缓存结构及容量比	缓存层次	访问次数	命中次数	命中率/%
传统结构	L1	1 000 000	722 311	72.23	渗透缓存 4 : 4	L1-draw	1 000 000	645 178	64.52
	L2	277 689	96 518	34.76		L1-push	354 822	46 194	13.02
						L2-draw	308 628	136 586	44.26
						L2-push	172 042	9762	5.67
	L3	181 171	61 534	33.97		L3-draw	162 280	82 175	50.64
整体缓存	1 000 000	880 363	88.04	L3-push		80 105	2 412	3.01	
渗透缓存 1 : 7	L1-draw	1 000 000	483 908	48.39		渗透缓存 5 : 3	整体渗透缓存	1 000 000	922 307
	L1-push	516 092	82 462	15.98	L1-draw		1 000 000	669 556	66.96
	L2-draw	433 630	141 162	32.55	L1-push		330 444	39 490	11.95
	L2-push	292 468	31 278	10.69	L2-draw		290 954	138 200	47.50
	L3-draw	261 190	93 559	35.82	L2-push		152 754	6945	4.55
	L3-push	167 631	6444	3.84	L3-draw		145 809	72 037	49.41
	整体渗透缓存	1 000 000	838 813	83.88	L3-push		73 772	1 803	2.44
渗透缓存 2 : 6	L1-draw	1 000 000	568 199	56.82	渗透缓存 6 : 2	整体渗透缓存	1 000 000	928 031	92.80
	L1-push	431 801	65 343	15.13		L1-draw	1 000 000	689 925	68.99
	L2-draw	366 458	134 272	36.64		L1-push	310 075	32 404	10.45
	L2-push	232 186	23 521	10.13		L2-draw	277 671	137 780	49.62
	L3-draw	208 665	94 142	45.12		L2-push	139 891	4852	3.47
	L3-push	114 523	4510	3.94		L3-draw	135 039	65 197	48.28
	整体渗透缓存	1 000 000	889 987	89.00		L3-push	69 842	1 195	1.71
渗透缓存 3 : 5	L1-draw	1 000 000	614 346	61.43	渗透缓存 7 : 1	整体渗透缓存	1 000 000	931 353	93.14
	L1-push	385 654	53 537	13.88		L1-draw	1 000 000	707 111	70.71
	L2-draw	332 117	133 965	40.34		L1-push	292 889	23 567	8.05
	L2-push	198 152	14 812	7.48		L2-draw	269 322	135 597	50.35
	L3-draw	183 340	92 268	50.33		L2-push	133 725	2 981	2.23
	L3-push	91 072	3 180	3.49		L3-draw	130 744	61 741	47.22
	整体渗透缓存	1 000 000	912 108	91.21		L3-push	69 003	685	0.99
					整体渗透缓存	1 000 000	931 682	93.17	

的时间及时局部性,并且从表2可以看出,随泉吸缓存容量和泉涌缓存容量的比逐渐增加时,处理器内核在泉吸缓存各层级的命中率保持稳步递增,而不是保持稳定不变或递减.因此,渗透缓存内泉吸缓存具备时间及时局部性.

4.2 测试程序 Cholesky

根据表3的实验数据,传统缓存结构的命中率明显低于渗透缓存结构.另外表3的数据表明,当泉吸缓存的容量和泉涌缓存的容量的比为1:7时,尽管访存命中大部分集中在泉吸缓存内,但是泉涌缓存第一级也出现了比较高的命中率,命中率达85.35%;随泉吸缓存容量和泉涌缓存容量的比值逐渐增加,这样

的在泉吸和泉涌缓存内的命中率分布基本维持不变,甚至整体在泉涌缓存内的命中率还略有提高.说明在泉吸缓存具备比较突出的时间及时局部性的同时,泉涌缓存(第一级)内存在比较明显的空间及时局部性.

4.3 测试程序 LU

根据表4,访存命中率大部分集中在泉涌缓存的第二级,并随泉吸缓存容量和泉涌缓存容量的比值逐渐增大,泉吸缓存第三级的命中率开始逐渐增加.实验结果说明,在该测试程序运行的过程中,泉涌缓存相对具有比较突出的空间及时局部性.另外根据表4的实验数据,传统缓存结构的命中率远远低于渗透缓存结

表3 Cholesky测试集缓存命中信息

缓存结构及容量比	缓存层次	访问次数	命中次数	命中率/%	缓存结构及容量比	缓存层次	访问次数	命中次数	命中率/%
传统结构	L1	1 000 000	899 480	89.95	渗透缓存 4 : 4	L1-draw	1 000 000	898 151	89.82
	L2	100 520	1 071	1.07		L1-push	101 849	97 873	96.10
						L2-draw	3976	2 223	55.91
						L2-push	1 753	243	13.86
	L3	99 449	521	0.52		L3-draw	1 510	555	36.75
整体缓存	1 000 000	901 072	90.11	L3-push		955	126	13.19	
渗透缓存 1 : 7	整体渗透缓存	1 000 000	999 171	99.92		L1-draw	1 000 000	898 700	89.87
					L1-push	101 300	97 778	96.52	
					L2-draw	3522	1 841	52.27	
					L2-push	1 681	238	14.16	
					L3-draw	1 443	481	33.33	
					L3-push	962	127	13.20	
					整体渗透缓存	1 000 000	999 165	99.92	
渗透缓存 2 : 6	整体渗透缓存	1 000 000	999 128	99.91	L1-draw	1 000 000	899 096	89.91	
					L1-push	100 904	97 668	96.79	
					L2-draw	3 236	1 621	50.09	
					L2-push	1 615	224	13.87	
					L3-draw	1 391	386	27.75	
					L3-push	1 005	133	13.23	
					整体渗透缓存	1 000 000	999 128	99.91	
渗透缓存 3 : 5	整体渗透缓存	1 000 000	999 137	99.91	L1-draw	1 000 000	899 372	89.94	
					L1-push	100 628	97 345	96.74	
					L2-draw	3 283	1 449	44.14	
					L2-push	1 834	187	10.20	
					L3-draw	1 647	317	19.25	
					L3-push	1 330	129	9.70	
					整体渗透缓存	1 000 000	998 799	99.88	

结构的命中率,推测可能是由于传统缓存结构中各级缓存的作用与渗透缓存中泉吸缓存的作用相当,当遇到更偏向空间局部性的程序时,需要泉涌缓存发挥更大的作用,而传统缓存结构仅存在“泉吸缓存”,因此传统缓存在访存命中率方面的性能大大下降,而渗透缓存可以通过增大泉涌缓存容量比,迎合程序的空间局部性,大大提高访存命中率。

仿真实验结果总体表明,运用渗透缓存比运用传统缓存提高了处理器内核的访存命中率,为缩短访存延迟营造出了及时局部性环境.不仅如此,处理器内核在渗透缓存不同区域成功访存结果将及时局部性的时间特征和空间特征区别开来.如果命中发生在泉吸缓存区域,那么说明该区域时间及时局部性特征相

对明显,访存命中率越高,其时间及时局部性越突出;如果命中发生在泉涌缓存区域,那么该区域空间及时局部性相对明显,访存命中率越高,其空间及时局部性特征越突出.这样的时间与空间及时局部性反应的正是处理器内核访问程序中指令和数据所展现的时间与空间及时局部性。

在 Wordcount 测试程序的实验中,无论泉吸缓存和泉涌缓存容量的配比如何,处理器内核访存命中区域均集中在第一级泉吸缓存、第二级泉吸缓存和第三级泉吸缓存,在泉涌缓存的命中相对少很多.比如,在泉吸缓存和泉涌缓存的配比为 1:7 时,第一级泉涌缓存为 15.98%,第二级泉涌缓存为 10.69%,第三级泉涌缓存为 3.84%.当泉吸缓存和泉涌缓存容量配比逐

表 4 LU 测试集缓存命中信息

缓存结构及容量比	缓存层次	访问次数	命中次数	命中率/%	缓存结构及容量比	缓存层次	访问次数	命中次数	命中率/%
传统结构	L1	877 441	7105	0.81	渗透缓存 4 : 4	L1-draw	877 441	3 206	0.37
	L2	870 336	15 500	1.78		L1-push	874 235	117	0.01
						L2-draw	874 118	7964	0.91
	L3	854 836	44 442	5.20		L2-push	866 154	770 055	88.91
						L3-draw	96 099	27 297	28.41
整体缓存	877 441	67 047	7.64	L3-push		68 802	3	0.00	
渗透缓存 1 : 7	L1-draw	877 441	1946	0.22	渗透缓存 5 : 3	整体渗透缓存	877 441	808 642	92.16
	L1-push	875 495	152	0.02		L1-draw	877 441	3912	0.45
	L2-draw	875 343	542	0.06		L1-push	873 529	104	0.01
	L2-push	874 801	805 888	92.12		L2-draw	873 425	12 148	1.39
	L3-draw	68 913	2071	3.01		L2-push	861 277	751 723	87.28
	L3-push	66 842	5	0.01		L3-draw	109 554	38 298	34.96
	整体渗透缓存	877 441	810 604	92.38		L3-push	71 256	3	0.00
渗透缓存 2 : 6	L1-draw	877 441	2 226	0.25	渗透缓存 6 : 2	整体渗透缓存	877 441	806 188	91.88
	L1-push	875 215	141	0.02		L1-draw	877 441	4847	0.55
	L2-draw	875 074	1 851	0.21		L1-push	872 594	90	0.01
	L2-push	873 223	795 990	91.16		L2-draw	872 504	16 661	1.91
	L3-draw	77 233	9017	11.68		L2-push	855 843	725 990	84.83
	L3-push	68 216	4	0.01		L3-draw	129 853	51 929	39.99
	整体渗透缓存	877 441	809 229	92.23		L3-push	77 924	3	0.00
渗透缓存 3 : 5	L1-draw	877 441	2 634	0.30	渗透缓存 7 : 1	整体渗透缓存	877 441	799 520	91.12
	L1-push	874 807	117	0.01		L1-draw	877 441	5953	0.68
	L2-draw	874 690	4249	0.49		L1-push	871 488	77	0.01
	L2-push	870 441	784 045	90.07		L2-draw	871 411	21 045	2.42
	L3-draw	86 396	18 450	21.36		L2-push	850 366	663 451	78.02
	L3-push	67 946	5	0.01		L3-draw	186 915	67 810	36.28
	整体渗透缓存	877 441	809 500	92.26		L3-push	119 105	3	0.00
					整体渗透缓存	877 441	758 339	86.43	

步增加条件下,这些在泉涌缓存发生的命中率还缓慢下降.因此,在处理器内核对测试程序 Wordcount 内的指令和数据的访问具有比较突出的时间及时局部性.此外,处理器内核对测试程序 Wordcount 的访问也具有一定程度的空间及时局部性,只是这种空间及时局部性相比起来不是十分突出.

在 Cholesky 测试程序实验中,无论泉吸缓存和泉涌缓存容量的配比如何,处理器内核对测试程序 Wordcount 内的指令和数据的访问具有比较突出的时间及时局部性.此外,处理器内核对测试程序 Wordcount 的访问也具有一定程度的空间及时局部性,只是这种空间及时局部性相比起来不是十分突出.

命中率呈大幅下降趋势,而在泉吸缓存第二级和第三级泉涌缓存的命中率几乎为零.因此,处理器内核对测试程序 Cholesky 内的指令和数据的访问同时具有时间及时局部性和空间及时局部性,且时间及时局部性稍微多一点.

在 LU 测试程序实验中,无论泉吸缓存和泉涌缓存容量的配比如何,处理器内核对测试程序 Wordcount 内的指令和数据的访问具有比较突出的时间及时局部性.此外,处理器内核对测试程序 Wordcount 的访问也具有一定程度的空间及时局部性,只是这种空间及时局部性相比起来不是十分突出.

仿真实验表明,从计算机内存迁移到片上缓存进入处理器内核的指令和数据确实具有时间与空间及时局部性,反映了这些指令和数据具有对处理器内核的访问而言的时间与空间及时局部性,它们是客观存在的,渗透缓存在处理器的片上营造出了容纳及时局部性的条件和环境,为提高处理器访存效率提供了一个新的系统机制。

5 总结

迁移到片上缓存的数据具有时间及时局部性和空间及时局部性,渗透缓存将这样的及时局部性清晰地展现出来,这种时间与空间的及时局部性同计算机的存储程序设计思想存在密切的渊源。渗透缓存作为新的片上缓存,进一步发挥了片上缓存在计算机体系结构上的特殊地位作用,凭借及时局部性这个认识分析计算性能的窗口,将时间与空间特性明确地展现了出来。

片上缓存的根本作用是空间换时间,在时间和空间转换的条件下,片上缓存的及时局部性的优势得到了进一步发挥,为提高处理器访存效率、减少处理器内核到内存里访问数据的时间消耗,缩短处理器内核的访存延迟创造了条件。

当指令和数据以及时局部组为单位整体地来到渗透缓存里,然后被化整为零布置在渗透缓存的各个层级之后,处理器内核访存命中率自然会得到提高,这是因为在指令和数据之间存在的时间和空间意义上的邻近关系增加了这些数据被访问的可能性,层级之间的分开也提高这些指令和数据被后续到来的指令和数据不被覆盖的可能性。指令和数据在渗透缓存上的如此分布形态,均能较好地适应处理器内核串行访存和跳转访存的行为特点。随处理器内核的访问逐步展开,指令和数据在渗透缓存中的迁移过程中逐渐形成指令或数据流,乃至涡流。指令和数据的流动自然促进了处理器内核访存的流畅性,计算性能得到改善。

渗透缓存的出现,从表象上来看,改变了片上缓存的结构,但本质上是改变了数据迁往片上缓存的策略和处理器内核访问片上缓存的方式,因为若将泉涌缓存和泉吸缓存的特点予以抽象,渗透缓存和传统缓存无本质区别,即改变原有渗透缓存中数据迁往片上缓存的策略和处理器内核访问片上的方式,把传统缓存的区域一分为二,仍能发挥渗透缓存带来的时间和空间及时局部性优势,但这些优势在当前传统的数据迁移、访问和缓存区域划分的设计策略下不能突出地显现出来。因此,从更一般的意义上讲,实现并保持时间和空间及时局部性不仅应该

成为优化片上缓存性能的探索方向,而且也应该成为未来高级处理器芯片研制所追求的目标。

参考文献

- [1] 胡九川, 范东睿, 李丹萍, 等. 一种支持数据渗透迁移的片上缓存模型研究[J]. 北京交通大学学报, 2017, 41(5): 1-9.
HU J C, FAN D R, LI D P, et al. An on-chip cache model research on supporting data permeation and migration[J]. Journal of Beijing Jiaotong University, 2017, 41(5): 1-9. (in Chinese)
- [2] TAN G M, SUN N H, GAO G R. Improving performance of dynamic programming via parallelism and locality on multicore architectures[J]. IEEE Transactions on Parallel and Distributed Systems, 2009, 20(2): 261-274.
- [3] GARCIA E, OROZCO D, KHAN R, et al. A dynamic schema to increase performance in many-core architectures through percolation operations[C]//20th Annual International Conference on High Performance Computing. Piscataway: IEEE, 2013: 276-285.
- [4] FAN D R, YUAN N, ZHANG J C, et al. Godson-T: An efficient many-core architecture for parallel program executions[J]. Journal of Computer Science and Technology, 2009, 24(6): 1061-1073.
- [5] 胡伟武, 陈云霁, 肖俊华, 等. 计算机体系结构[M]. 北京: 清华大学出版社, 2011: 185-202.
HU W W, CHEN Y Q, XIAO J H, et al. Computer Architecture[M]. Beijing: Tsinghua University Press, 2011: 185-202. (in Chinese)
- [6] PATTERSON D A, HENNESSY J L. Computer Organization and Design: The Hardware/Software Interface[M]. 5th Edition. Saint Louis: Morgan Kaufmann, 2014: 280-305.
- [7] KIM J, JANG H, LEE H J, et al. UC-check: Characterizing micro-operation caches in x86 processors and implications in security and performance[C]//54th Annual IEEE/ACM International Symposium on Microarchitecture. New York: ACM, 2021: 550-564.
- [8] JIANG S Z, YANG Q S, CI Y W. Merging similar patterns for hardware prefetching[C]//2022 55th IEEE/ACM International Symposium on Microarchitecture (MICRO). Piscataway: IEEE, 2022: 1012-1026.
- [9] BAKHSHALIPOUR M, SHAKERINAVA M, LOTFI-KAMRAN P, et al. Bingo spatial data prefetcher[C]//2019 IEEE International Symposium on High Performance Computer Architecture (HPCA). Piscataway: IEEE, 2019:

399-411.

- [10] WANG Z R, LIU C, BECKMANN N, et al. Affinity alloc: Taming not-so near-data computing[C]//56th Annual IEEE/ACM International Symposium on Microarchitecture. New York: ACM, 2023: 784-799.

作者简介



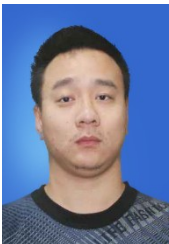
胡九川 男,1965年出生,重庆人.北京交通大学副教授、硕士生导师.主要研究方向为计算机体系结构、软件工程等.

E-mail: jchhu@bjtu.edu.cn



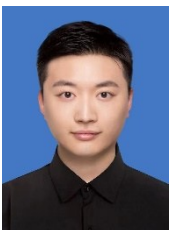
程建聪 男,1997年出生,山西运城人.教育部教育管理信息中心,工程师.主要研究方向为计算机体系结构.

E-mail: chengjiancong@moe.edu.cn



万良易 男,1995年出生,江西九江人.中国电子科技集团公司第32研究所工程师.主要研究方向为计算机体系结构.

E-mail: wly950505@163.com



吴楠士 男,1995年出生,内蒙古自治区赤峰人.北京交通大学硕士生.主要研究方向为计算机体系结构.

E-mail: 20140087@bjtu.edu.cn



叶笑春 男,1981年出生,江西人.中国科学院计算技术研究所副研究员、硕士生导师.主要研究方向为众核处理器体系结构、高性能计算、高通量计算、软件模拟技术等.

E-mail: yexiaochun@ict.ac.cn



严龙 男,1988年出生,北京人.中国科学院计算技术研究所,工程师.主要研究方向高通量计算机体系结构.

E-mail: yanlong@ict.ac.cn