

Page Size Aware Cache Prefetching 研读报告

- 姓名：郭裕彬
- 学号：2114052
- 专业：物联网工程

研究问题

随着机器性能的不不断提升，当代应用程序工作集的大小增长超过了缓存大小的增长，导致了频繁的主存访问，这使得程序需要花费大量时间，恶化了系统的性能。“预取”在需求访问出现之前通过预测手段主动将对应数据块预取到缓存层次结构中，能够有效地缓解大型工作集的应用对内存子系统施加的压力。

其中，空间预取器相比于时间预取器有着较大的优势，但它们假定只使用标准的4KB物理页面，将模式检测限制在4KB内存区域内。由于虚拟地址连续的内存块自物理地址空间中映射来的物理内存块不一定是连续的，因此这些空间缓存预取器不允许超过4KB物理页边界的预取；此外，跨越4KB边界可能会引入一些访问非法地址的安全漏洞，攻击者容易使用页面交叉预取来攻击系统。在这种限制下，若要启用超过4KB的安全预取，需要直接访问TLB。对于在物理地址空间中操作的空间预取器来说，从低级缓存直接访问TLB并进行多值函数性质的反向映射是一个十分巨大的开销，这使得频繁使用大于4KB的页面的现代系统并没有因为这种预取器而显著提高性能。

主要贡献

- 现代操作系统和体系结构针对小而快且漏报率低的TLB的需求，提供了对不同页面大小的支持，流行的X86架构已经能够支持1GB的大页。本文针对这个变动，通过实验测试揭示了借助大页面的支持可以提高原本空间缓存预取器在物理地址空间中操作的有效性。
- 本文提出了页面大小传播模块(PPM)，作为第一个允许超过4KB物理页面边界的安全预取的微架构方案。将这个方案与当前的空间缓存预取器SPP、VDLP、PPF和BOP相耦合，能够有2%至5%不等的单核速度提升。此外，PPM 无需修改设计即可与任何缓存预取器兼容。
- 本文利用PPM，将大页面透明地集成到任意预取器的实现中，并设计了一个复合的预取器Page Size Aware prefetcher with Set Dueling(Pref-PSA-SD)选择不同大小的Prdf-PSA来实现预取。单核评估表明上述的空间缓存预取器复合后在80种工作负载种相比原始版本的性能提高了2%至8%不等，多核环境下，几何均值加速比高达7.7%。

主要优点

- 拟议的页面大小传播模块 (PPM) 可实现超过 4KB 物理页面边界的安全预取，从而提高空间缓存预取器的有效性。观察到现代系统在执行内存密集型应用程序时大量使用2MB页面后，作者通过多种评估模型的测试发现固定使用2MB页面对性能的影响是不确定的，取决于工作负载。而采取动态方式感知页面大小，使用一种将页面大小信息传递给较低级缓存预取器的方案和一种确保这样做会对性能产生积极影响时才使用2MB页面的智能机制相配合，则具备不错的性能提升潜力。
- 使用PPM增强的缓存层次结构通过一个额外的位来表示响应访问块的页面大小，从而增加缓存MSHRs，这避免了对底层预取器实现的修改，以及反向虚拟地址到物理地址的转换。通过在上一级缓存缺失时更新这个额外的位，预取器得知能否安全访问4KB更大的空间，进而实现页面的动态调整。此时，预取器仍然是使用4KB的页面对内部预取结构进行索引，不会更改其设计。

- 本文通过PPM设计的复合预取器，对于4KB&2MB设计，一个预取器固有地使用4KB的页面(Pref-PSA)，另一个预取器固有地使用2MB的页面，采用set-dueling机制（将两种技术应用到不同的两个cache set上，然后统计两个set上的运行情况，然后来决断到底使用两种技术中的哪一个，然后将该算法策略部署到其余各个set上）来实现自适应选择，这是一种基于训练的方案，在实验中发现三个比特位的set就足以动态识别每个执行阶段最有用的缓存预取器。

主要缺点

- 随着CPU核数的增加，本文提出的PPM与空间预取器组合得到的性能提升变低，这是由于核心数增加与其他硬件设置的增量并不成比例，带宽成为了影响大小页面预取的更重要因素。
- PPM通过一个额外的位来实现两种大小页面的选择，但当页面的分级变多、大页的容量上升到GB量级时，所需要的额外位数会增加，这也就意味着更加复杂的解析架构，一定程度上会影响性能的提升。

改进思路

- 评估使用PMM技术对系统功耗和能效的潜在影响，因为这些因素在现代计算系统中至关重要。
- 调查PMM设计在大型四核、八核甚至更多核系统中的可扩展性，并评估其不同内存密集型场景下的性能。