**海量存储课程**

**利用延迟变化来减少NAND闪存的访问冲突**

**(主要文献来源MSST-2016)**

院（系）名称：计算机学院

专业名称：计算机技术

学生学号：2017282110240

学生姓名：夏晨

摘要

自当今社会进入互联网、大数据时代以来，对大容量存储技术提出了巨大需求。 NAND Flash特别是一个存储单元存储 2 位数据的 MLC NAND Flash ，由于其具有功耗小、大容量、体积小、随机读写性能好、高性价比等优点得到了极大应用。

最近，耐久性，写入速度和读取速度之间的权衡在许多方面已经被用于I/O性能改进，这也引起读/写延迟变化。在本文中，延迟变化被利用在I/O调度访问特性引导读写延迟最小化。

首先，在了解读取延迟，写入延迟和原始误码率（RBER）之间的关系的基础上，讨论了利用读写延迟减少之间关系的不同方法。然后，利用访问数据的热度和滞留时间来确定写入或读取的速度，提出了一种I/O调度方案，为快速写入和快速读取提供调度优先级以减少冲突。各种迹象的实验表明，所提出的技术实现了显着的读写性能改进。

**关键词：**NAND；读写性能；访问冲突;FIFO

目录

[1 绪论 1](#_Toc27106)

[1.1 研究背景 1](#_Toc32689)

[1.2 研究现状 1](#_Toc12285)

[1.3 本论文研究内容与意义 2](#_Toc21468)

[1.4 本论文的组成 2](#_Toc27686)

[2 背景和相关工作 3](#_Toc28930)

[2.1 RBER，写入速度和读取速度之间的权衡 3](#_Toc11508)

[2.2 相关工作与技术 4](#_Toc16000)

[2.2.1 过程变化 4](#_Toc2787)

[2.2.2 滞留时间变化 4](#_Toc28434)

[2.3 冷热数据与强弱块 5](#_Toc16689)

[2.4 本章小节总结 6](#_Toc32451)

[3 滞留感知和热冷感知I/O调度算法 7](#_Toc23153)

[3.1 算法基本思想 7](#_Toc14759)

[3.2 热冷感知写入调度 7](#_Toc30452)

[3.2.1 热冷感知写入调度技术分析 7](#_Toc22834)

[3.2.2 实现流程与方法 8](#_Toc9014)

[3.3 滞留感知的读调度 9](#_Toc30049)

[3.3.1 滞留感知的读调度技术分析 9](#_Toc9965)

[3.3.2 实现流程方法 9](#_Toc12306)

[4 实验与分析 11](#_Toc14711)

[4.1 实验设置 11](#_Toc29760)

[4.2 实验结果分析 11](#_Toc11949)

[5 结论 14](#_Toc12286)

2. 绪论
   1. 研究背景

基于NAND闪存的固态驱动器（SSD）与硬盘驱动器（HDD）相比，具有更高的随机存取性能，更低的功耗，抗冲击性以及无噪声的优点，在数据中心和个人计算机中已经获得了广泛的普及。在过去十年中，由于技术从65纳米技术扩展到最新的10纳米技术，NAND闪存的容量不断增加，比特密度从每个单元1位增加到最新的每个单元6位。不幸的是，随着闪存密度的增加，NAND闪存的可靠性由于包括滞留噪声，读取干扰噪声，单元间编程干扰噪声和编程/擦除（P/E）循环噪声等多种错误源而显着降低，这带来了新的挑战。一方面，写入速度降低，以抵消寿命降低和原始误码率（RBER）增长。另一方面，更复杂的纠错码（ECC）方案如低密度奇偶校验码（LDPC），所驱动的更强的纠错能力将显着增加访问延迟（读/写/擦除操作）。因此，寻求改善闪存I/O性能的方法是有动机的。

* 1. **研究现状**

考虑到RBER，写入速度和读取速度之间的权衡，一种有前景的I/O性能改进技术。闪存读取速度与闪存单元的RBER高度相关。RBER越高，ECC能力越强，ECC方案的复杂度越高，读请求越慢。另外，RBER和写操作速度之间有着密切的关系。有几项工作指出，使用递增阶跃脉冲编程（ISPP）方案的较小程序步长ΔVp（其逐渐累积电压以超过阈值电压）会以写入速度降低为代价降低RBER。因此，在低RBER，快速读取和快速写入这三个目标中，任何两个都可以以牺牲第三个为代价来优化。

具体来说，在闪存中观察到日益显着的过程变化（PV），这使得这种权衡可以针对闪存中的不同块进行单独调节。PV是在相同的P/E循环下，不同存储器块中的页面的最坏情况RBER可能存在巨大差别，这相当于当给定相同的ECC时，不同存储器块内的P/E循环耐久性差异巨大。这是由氧化层厚度和栅极宽度/长度的显着变化引起的，随着技术规模的不断扩大，因此，在进行所论述的权衡方案时，可以假定最坏情况下的块行为或针对不同块强度单独地优化。

到目前为止，已经投入了很多的努力来尝试利用权衡来加速性能，这可以分为三类。第一组工作是关于利用PV(过程变化)。迄今为止唯一的现有工作由Shi等人提出。使用粗糙的ΔVp的强健页面，这些页面不会像以前那样累积错误，并为其分配热门数据。第二组是根据数据滞留时间---自闪存单元编程以来的时间长度来调整单元编程/读出参数。当数据实际时间增加时，可以用较低的读出阈值，而当估计的滞留时间较高时，可以使用较高的编程电压或更精细的ΔVp另一组是识别只读/只写页面，使得读写操作可以相互加速，而彼此间的影响更小。当利用所论述的权衡方案来改善I/O性能时，请求仅被部分加速，诸如读/写热请求，这不可避免地导致显着的读写速度变化，并且因此促使搜索方法利用它来进一步优化。

* 1. **本论文研究内容与意义**

在本文中，我们提出了一种用于NAND闪存的滞留感知和冷热感知的I/O调度算法（RHIO）。我们的主要观点是：通过赋予快速写入和快速读取的调度优先级，可以利用由权衡感知技术引起的速度变化来获得最大的益处。首先，为了放大强健块所带来的好处，热写入被优先调度，其中数据被分配给较低写延迟的较强健块。其次，根据访问数据的实际滞留时间对读请求进行排序，并且优先对低读延迟的热读取进行排序。因此，使用RHIO可以通过使用三种技术来实现显着的I/O性能改进，包括PV感知写入速度调节，滞留感知读取速度调节和最短作业调度。  
我们通过代表性的工作负载和基于跟踪的仿真来评估算法的有效性，结果表明，平均来说，RHIO能够显着提高读写请求的性能39.11％和29.92％。此外，通过使用RHIO调度程序收集优先级写入和读取请求的百分比，结果清楚地表明了RHIO在减少I/O延迟方面的有效性，尤其是对于读写密集型应用程序

* 1. **本论文的组成**

本文的其余部分安排如下。第二节介绍背景和相关工作。第三节描述了我们的闪存设备I/O调度方案的设计技术和实现问题。第四节介绍了实验和结果分析。第五部分总结了本文的研究结果。

1. 背景和相关工作

在本节中，我们首先要更好地理解RBER，读取速度和写入速度之间的权衡。然后，介绍了与此权衡有关的以前的研究，以进一步在这方面的工作。

* 1. RBER，写入速度和读取速度之间的权衡

RBER，写入速度和读取速度之间的权衡取决于两个关系：

1. ECC复杂性，纠错能力和读取速度。
2. RBER，程序步长和写入速度。

在这项工作中，我们将LDPC作为默认的ECC方案，同时带来更好的纠错能力以及读取性能下降。

第一种关系是由于软判决记忆感知，它使用两个相邻存储状态之间的多个量化等级。一方面，随着在两个相邻存储状态之间使用的量化等级的数目增加，旨在感测和数字量化每个存储器单元的阈值电压的读取操作被延迟。另一方面，感测电平的数量也影响LDPC码解码的纠错强度。更多的感应电平意味着在NAND闪存的背景下进行精确的存储感应，从而为LDPC码解码提供更准确的每个比特的输入概率信息，从而提高其纠错能力。因此，可以探索纠错能力和读取速度之间的权衡。

基于RBER应该在部署的LDPC码的纠错能力范围内的前提条件，从上述两个关系可以推断RBER，读取速度和写入速度之间的权衡。对于RBER较低的块，可以改进读取或写入速度，反之亦然。在这项工作中，我们关注由这种权衡引起的读写速度变化。

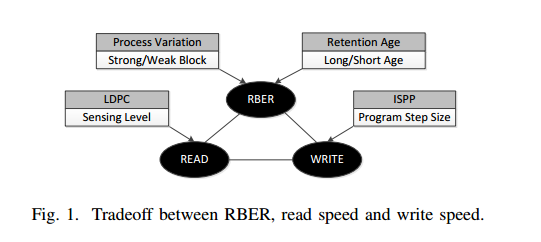


图2.1 RBER、读取速度和写入速度之间的权衡。

 已经提出了几种用于改进I/O性能的方法来利用RBER，读取速度和写入速度三者之间的权衡。包括充分利用PV引起的强健块，考虑滞留时间短的数据，根据访问特性调整程序步长和感知水平。图2.1显示了利用权衡的多种方式。就我们所知，这是利用RBER，写入速度和读取速度之间的权衡来首次呈现这种图表。我们相信系统设计人员对于闪存I/O性能和耐用性的进一步优化将是有启发的。

* 1. 相关工作与技术
     1. 过程变化

过程变化：PV是在相同的P/E循环下，不同存储器块中的页面的最坏情况RBER可能存在巨大差别，这相当于当给定相同的ECC时，不同存储器块内的P/E循环耐久性差异巨大。

以前的工作表明闪存块的RBER往往遵循对数高斯分布。PV导致具有更强和更弱的单元的块，其分别累积比特误差更慢或更快。在这项工作中，PV是通过周期性地读取块来测量的，并找出ECC需要纠正多少位。

最近，利用PV的技术主要集中在损耗平衡方面，因为它充分利用固态硬盘中的强健的块来最大化使用寿命。例如，潘等人。通过使用RBER统计作为耗损均衡算法的存储器块磨损速度的测量来延长闪存寿命。Woo等人引入了一种新的测量方法，它比擦除计数更准确地预测闪存块的剩余寿命，其基础是所有闪存块都能够存活得比保证数量长得多，并且P/E周期数目显着地变化块。据作者所知，Shi等人提出的PV感知数据分配方法是唯一一个同时考虑PV和RBER与写入速度之间的权衡，使用较粗的ΔVp作为较快的页面，而不是以较快的速度累积错误，并以较热的数据与较快的数据块匹配的方式分配块。在本文中，我们的热冷感知写入调度算法还利用基于PV感知数据分配的强健块。

* + 1. 滞留时间变化

滞留时间的变化：数据滞留时间是自闪存单元编程以来的时间长度。随着时间的推移，由浮栅泄漏的电荷引起的数据保持误差是主要的误差之一。因此，滞留时间变化会导致不同的RBER，这可以像PV感知技术一样被利用

已经彻底分析了数据滞留时长对存储系统性能的影响。有些作品专注于最小化刷新成本。例如，罗等人。引入了针对NAND闪存的称为WARM的写入热感知的滞留管理策略，该策略允许闪存控制器通过利用该数据的高写入频率来放宽写入数据的闪存滞留时间，而不需要刷新。最近Di等人。提出了一种刷新最小化方法，将长滞留时间要求的数据写入高耐久性块。另一组方法适应单元编程/读出参数以提高性能。例如，Cai等人文献提出了一种滞留优化读取（ROR）方法，该方法周期性地学习一个紧密的上限，并及时的为每个闪存块提供最佳读取参考电压。施等人。当滞留时间较短时，通过降低编程电压，提出了一种滞留修整方法来减少磨损。刘等人通过适应编程步长ΔVp和ECC强度，基于估计的滞留时间实现了写入响应时间加速。这些研究表明，在工作负载中滞留时间的变化非常明显且有用。在本文中，我们的滞留感知读取调度算法利用基于滞留感知的ECC适配得到了滞留时间较短的数据的优点。

* 1. 冷热数据与强弱块

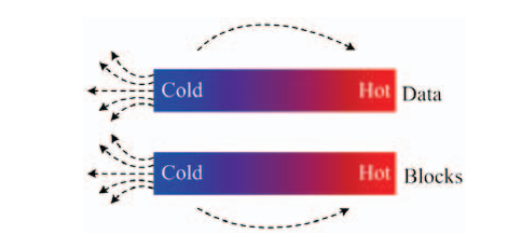


图2.2 冷热数据

需要较短保留时间的数据将被频繁更新，称为热数据，保留时间较长的数据称为冷数据。

我们假设在早期检测期间和之后的这种非单调的错误率行为可以由闪存块的两个池模型来解释：一个块池称为较弱的池，其由错误率增加得快得多的单元比另一个街区，称为更强大的游泳池。较弱的池快速产生不可纠正的错误（导致在早期检测期间观察到的失效率增加，因为这些块保持失败）。包含此池的单元最终会失败，并由SSD控制器尽早停止使用。随着较弱池中的块被耗尽，整体错误率开始下降（正如我们在所谓的早期检测阶段结束之后所观察到的那样），并且它继续下降，直到较强的池中更耐久的块开始耗尽由于典型的使用。

* 1. 本章小节总结

当使用基于PV的快速写入和滞留时间为基础的快速读取时，请求以不同程度加速，这不可避免地导致显着的读写速度变化。幸运的是，I/O调度程序是利用速度变化来提高读写性能的一个很好的选择。虽然大多数基于闪存的I/O调度器侧重于如何通过利用SSD的内部并行机制来减少访问冲突和提高芯片利用率，但我们关注的是减少冲突时的访问冲突延迟不可避免地，通过利用速度变化。

1. 滞留感知和热冷感知I/O调度算法

在本节中，提出了滞留感知和热冷感知I/O调度算法（RHIO）。RHIO的设计原则是基于从我们的数据集中得出的观察结果和分析：如果权衡感知技术根据属性的变化特征来提高I/O性能，那么属性的检测可以在I/O调度中实现，因此权衡引起的速度变化可以通过给予快速写入和快速读取的调度优先级来利用最大的益处。

* 1. 算法基本思想

RHIO的基本思想是将写入请求分为不同的队列（取决于其热度），而读取请求则根据访问数据的滞留时间分开。对于热写入，使用快速写入将其数据分配给强健块，并给予调度优先级。对于访问滞留时间短的数据的读取，优先执行快速读取和调度，以最小化I/O请求的访问冲突延迟。

* 1. 热冷感知写入调度
     1. 热冷感知写入调度技术分析

通常，写入请求的访问延迟由访问冲突延迟，数据传送延迟和程序延迟组成。与读取请求相比，写入请求更加耗时，导致严重的访问冲突延迟。感知热度的写入调度方案旨在通过利用PV引起的RBER变化，权衡引起的写入速度变化和访问特性引起的热度变化来减少访问冲突延迟。

在所提出的热冷感知写入调度方案中，我们使用快速写入将强热数据置于强热区块中，将非热数据置入正常写入的正常区块中。强健块的热数据很快就会失效，从每个P/E周期具有较低RBER增长率的强健区块中受益最大。此外，优先发出热写请求以减少队列中的下几个请求的冲突延迟。请注意，使用不同的热/冷分类方案与我们的方案的设计是正交的。在这项工作中，我们根据其文件系统I/O请求的大小来识别热/冷数据，这在以前的几项研究中已被证明是简单而有效的。有人建议，请求大小越大，数据越冷。采用热组作为主数据结构，将相同热请求调度到同一个组中。

所提出的方法可以保证高热度写入请求不被低热度写入请求阻塞。但是，这可能会导致低热请求长时间在队列中等待，而无需服务。为了避免冷请求的潜在饥饿，每个传入请求被分配一个截止时间，该截止时间定义了发出请求的最晚时间点。在串行ATA（SATA）接口中，通过在NCQ命令（读取FPDMA排队和写入FPDMA排队）的优先级（PRIO）字段中放置‘01b'来配置截止日期信息，将其标记为等时，然后在等时命令完成（ICC）字段中填写相应的截止时间值。请注意，ICC位7被清除为零，因此时间间隔是10毫秒的细粒度。先进先出（FIFO）队列按照到达时间顺序将请求连接在一起，被RHIO采用并定期检查。当FIFO队列的头部请求达到预设的截止时间时，该请求不再被阻止，并且将被立即处理以确保热度优先策略不会导致潜在的饥饿。

* + 1. 实现流程与方法

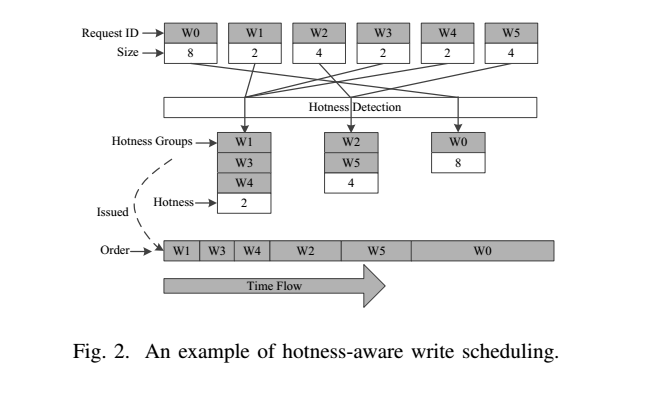


图3.1热冷感知写入调度的一个例子

图3.1显示了一个热冷感知写入调度的例子。如图所示，访问相同芯片的六个写请求按照FIFO顺序从左到右被添加到写请求队列，其中Size（W0）=8，Size（W1，W3，W4）=2，Size（W2，W5）=4。基于热度检测，创建三个热度组，其中HG1={W1，W3，W4}，HG2={W2，W5}，HG3={W0}。最后，热冷感知写入调度方案发出热度组中的I/O请求，其中所有热度组以热度顺序处理。通过将PL（大小）表示为对于具有大小页面的请求写入一页的等待时间，将N表示为冲突中记录的请求的数量，将Oi表示为请求i的发出顺序，则平均请求响应时间可以被定义为：

 (1)

在这个例子中，我们假设PL（2）=200μs，PL（4）=220μs，PL（8）=240μs，算法的平均请求响应时间为2053.3μs，正常算法的平均请求响应时间为3320.0微秒。因此，在热冷感知写入调度方案中，通过优先调度访问强健块的高热度的请求，这一点充分利用写入速度变化。

* 1. 滞留感知的读调度
     1. 滞留感知的读调度技术分析

与写入请求类似，读取请求的访问延迟由访问冲突延迟，数据传输延迟和感知延迟组成。由于整体系统性能很大一部分与存储平均读取响应时间有关，因此读取请求的性能是至关重要的。因此，读取和写入分离通常用于优先化读取请求的调度，这保证了读取请求不会被耗时的写入请求所阻塞。在这项工作中，滞留感知读调度方案旨在通过利用权衡导致的读取速度变化和访问特性引起的数据滞留变化来减少读冲突延迟。

在滞留感知读取调度方案中，读取请求根据要读取的数据的滞留时间进行排序。对于访问滞留时间短的数据的读取，优先执行快速读取和调度，以最小化I/O请求的访问冲突延迟。滞留时间的确定可以通过对FTL中每个映射条目的扩展字段----时间戳字段，记录数据被编程时的时间戳来实现。与写入调度中的基于大小的热度检测不同，读取调度中的热度检测是基于滞留时间,是实际时间而不是预测时间。请注意，由于PV在写入时已经被利用，所以我们只考虑RBER累积作为数据滞留时间的函数，而没有考虑块的强健度。

该方案在主机接口逻辑（HIL）中实施，其中，flash translation layer层（FTL）中所记录的SSD特定特性和数据编程时间戳的有关信息有助于针对I/O请求，实现特定设备更好的调度决策。注意，在热冷感知写入调度方案中提出的FIFO队列也起到防止读取饥饿的发生的作用。

* + 1. 实现流程方法

图3.2显示了滞留感知读取调度的示例。如图所示，访问同一芯片的六个读请求按照FIFO顺序从左到右被添加到读请求队列中。根据滞留检测，创建红黑树。由于R3所访问的数据的时间戳最大，这意味着数据是最新的，所以R3是第一个要发布的请求。

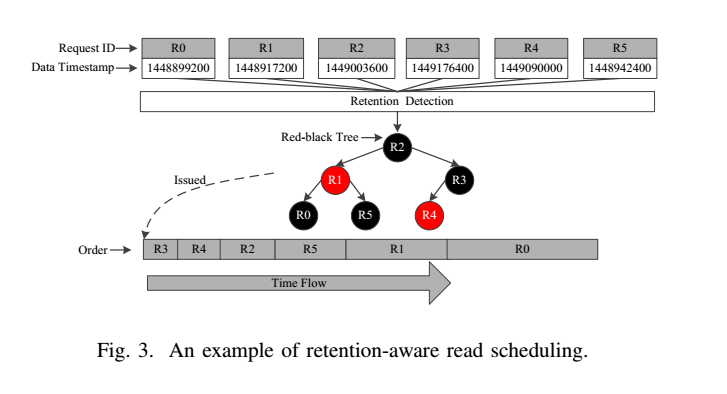


图3.2滞留感知读取调度的一个例子

通过将热冷感知写入调度和滞留感知与读取调度相结合，RHIO的实现需要在I/O队列中维护一个热度组列表和一个红黑树。由于I/O调度器的队列长度是有限的，所以存储成本可以忽略不计。所提出的方法的主要计算开销，这包括通过为写入请求寻找适当的热度组并将读取请求插入到红黑树中而引起的时间开销，其中复杂度与队列长度的对数成比例。另外，采用SSD内部并行机制的最先进的I/O调度算法与我们的工作有些正交。

1. 实验与分析
   1. 实验设置

在本文中，我们使用一个事件驱动的模拟器来进一步论证所提议的RHIO的有效性。我们模拟了8个通道的128GBSSD，每个通道连接8个闪存芯片。采用有界高斯分布的RBER增长率s来模拟闪存的过程变化，其中平均值μ和标准偏差σ分别设为3.7×10-4和9×10-5。当ΔVp为0.3时，我们使用600μs作为2位/单元NAND闪存编程延迟，当使用LDPC和7个参考电压时，存储器感测延迟为90μs，数据传输延迟为80μs。所有这些设置都与以前的工作实验一致。

为了进行验证，我们实施了RHIO以及基线NOOP调度，PV-W和RT-R。PV-W实现PVaware写入性能改进，而不需要冲突感知的重新排序，而RT-R在不重新排序I/O请求序列的情况下实现保留感知读取性能改进。我们使用来自MSRCambridgetrace的实际工作负载来评估我们的设计，这在以前的作品中被广泛用于研究SSD性能

* 1. 实验结果分析

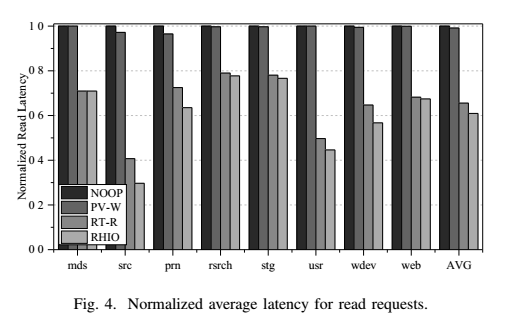


图4.1读取请求的标准化平均延迟

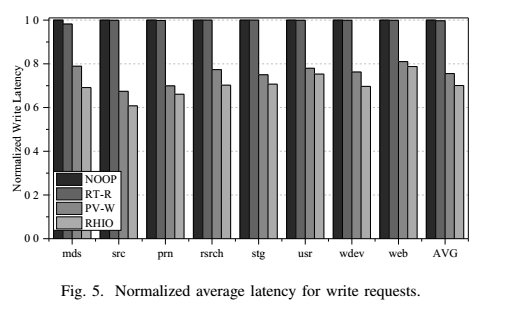


图4.2写入请求的标准化平均延迟

在这部分，实验结果被提出和分析。读写延迟通常用于评估调度性能。图4.1和图4.2分别显示了读写请求的标准化平均延迟。与传统的NOOP相比，RHIO实现了显着的读写性能提升。对于读取请求，受益于滞留引起的读取速度变化和冲突感知重新排序，RHIO平均减少读取延迟39.11％。对于写入请求，RHIO通过给热写入请求提供调度优先级，并将其数据分配给快速写入的强健块，从而实现了29.92％的写入延迟减少。另外，RHIO的性能优于RT-R和PV-W，读取延迟分别平均降低7.04％和38.56％。这是因为滞留感知读取调度方案通过优先发出快速读取请求来减少冲突延迟。与读取请求类似，通过利用快速写入请求来减少冲突，RHIO与RT-R和PV-W相比分别实现了29.71％和7.12％的写入延迟。

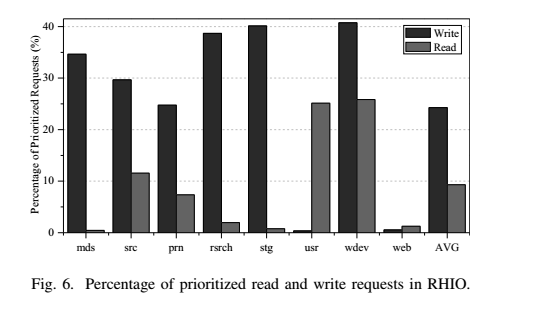


图4.3 RHIO中优先级读取和写入请求的百分比

但是，对于不同的跟踪，读写性能的改善是非常不同的。例如，与PV-W相比，在MDS中观察到的最大写延迟减少是12.42％，而在网中观察到的最小减少仅为2.82％。与RT-R相比，在src中观察到的最大读取延迟减少是27.13％，而在MDS中观察到的最小减少仅为0.1％。为了理解跟踪中不同性能改善的原因，使用RHIO调度程序收集的优先级写入请求和读取请求的百分比如图4.3所示。通过比较结果和图4.2所示的写入性能改进，可以观察到当优先级写入比率较高时，与PV-W相比，写入延迟减少较大。例如，优先写入比率大于25％的大多数跟踪的平均写入延迟显着降低，而相比于PV-W，在RHIO中优先写入仅为0.34％和0.56％的usr和web的写入延迟略微分别下降了3.34％和2.82％。这是因为在一些跟踪中更密集的I/O会导致更多的冲突和更长的冲突延迟，而在我们提出的RHIO中，通过调度更多优先级更高的写请求，这种情况已经减少了。阅读结果也支持读取结果。大多数应用程序的优先读取请求的百分比在5％以内，而对于src，prn，usr和wdev，优先读取请求的百分比超过5％，与RT-R相比读取性能提高。总的来说，这些结果清楚地表明了RHIO在降低读写延迟方面的有效性，特别是读写密集型应用。

1. 结论

在本文中，我们已经提出了基于NAND闪存的SSD的滞留感知和热度感知I/O调度算法（RHIO）。与以前的作品不同的是，块之间的延迟变化被用来指导读写延迟最小化。RHIO设计背后的关键洞察是，一个热冷感知写入调度方案可以通过给热写入请求调度优先级并以快速写入将其数据分配给强健块来减少写入冲突延迟，而滞留感知读取调度方案可以通过使用快速读取来优先调度以低滞留时间访问数据的读取请求来减少读取冲突延迟。广泛的实验结果和详细的比较表明，所提出的技术在读取和写入请求方面实现了显着的性能提高，平均提高了39.11％和29.92％