第 11 卷 第 3 期
2022 年 5 月
集 成 技 术
JOURNAL OF INTEGRATION TECHNOLOGY
Vol. 11 No. 3
May 2022

引文格式：
罗龙飞, 李蓍城, 石亮. 消费级混合式固态存储分析与研究 [J]. 集成技术, 2022, 11(3): 71-84.
Luo LF, Li SC, Shi L. Performance analysis and study for hybrid NAND flash memory [J]. Journal of Integration Technology, 2022, 11(3): 71-84.

消费级混合式固态存储分析与研究

罗龙飞# 李蓍城# 石 亮*

（华东师范大学计算机科学与技术学院 上海 200062）

摘 要 混合式固态存储已成为当前消费级终端领域的主流存储设备。然而在学术领域，关于混合式固态存储设计和问题的讨论与分析仍不够充分。该文针对现有的混合式存储设备，结合相关领域前沿研究，从混合式闪存架构介绍、亟待解决的痛点问题和相关研究进展 3 个方面进行讨论和分析。文章介绍和分析了混合式闪存的主流架构及其特点，展示了在真实设备平台上测试的实验数据结果，揭露了混合式闪存中亟待解决的问题，重点介绍了读特征、写特征、读写冲突和容量特征相关问题。同时介绍了相应问题的最新研究进展，并分析了各个技术的优劣和未来的发展方向。

关键词 混合式闪存；读写冲突；读特征；写特征；容量特征
中图分类号 TP 333.7 **文献标志码** A doi: 10.12146/j.issn.2095-3135.20220225001

Performance Analysis and Study for Hybrid NAND Flash Memory

LUO Longfei# LI Shicheng# SHI Liang*

(*Computer Science and Technology*, *Eastt China Normal University*, *Shanghai* 200062, *China*)

**Corresponding Author*: *lshi@cs.ecnu.edu.cn*

#Equal Contribution

Abstract Hybrid flash storage has become the mainstream storage device in the field of consumer device. However, the academic study on hybrid flash storage is still insufficient. Based on our research activities, practical experience on hybrid storage devices, and state-of-the-art researches, this paper introduces the architecture of hybrid flash memory, the pain points that need to be solved and the relevant research progress. Firstly, this paper introduces and analyses the hybrid flash memory architecture and the corresponding characteristics. Then the experimental results on real hybrid flash memory are shown and the problems of the hybrid flash memory to be solved are exposed. These problems are full into four categories, write characteristics, read characteristics, read/write interference, and volume characteristics. Finally, the

收稿日期：2022-02-25 **修回日期**：2022-04-07
基金项目：国家自然科学基金面上项目(62072177)；上海市自然科学基金面上项目(20ZR1417200)
作者简介：罗龙飞(共同第一作者)，博士，研究方向为操作系统和闪存存储；李蓍城(共同第一作者)，硕士，研究方向为操作系统和闪存存储；石亮(通讯作者)，教授，研究方向为计算机系统，E-mail：lshi@cs.ecnu.edu.cn。

latest research progresses of the corresponding problems are introduced. The advantages and disadvantages of each technique are summarized. Additionally, the future development direction is commented.

Keywords hybrid flash memory; read/write interference; read characteristics; write characteristics; volume characteristics

Funding This work is supported by National Natural Science Foundation of China (62072177), and Natural Science Foundation of Shanghai (20ZR1417200)

1 引 言

随着消费级终端的逐渐普及，闪存存储设备步入了快速发展阶段。相较于传统的硬盘，闪存存储设备功耗低、抗震强、性能更佳，可以满足用户的日常需求。相较于其他非易失性存储，闪存存储设备具有更大容量、更低价格和更优性价比。因此，闪存存储设备已成为消费级终端的主流存储设备。

闪存存储设备内部由通道、芯片、晶元、平面、块和页面组成，同时闪存具有写前擦除、擦除粒度与访问粒度不一致和擦除次数有限的特点。闪存转换层(Flash Translation Layer，FTL)的提出有助于更好地使用闪存存储设备。FTL 的主要功能包括地址映射、垃圾回收和磨损均衡等。

(1)地址映射：地址映射是为了将主机端发送的逻辑地址转换成实际存储在闪存存储设备中的物理地址。映射表为主要的元数据，维护了逻辑地址到物理地址的转换和一些页面的存储状态标识。

(2)垃圾回收：闪存具有写前擦除的特性，且擦除粒度大于访问粒度，因此为避免频繁的擦除操作，数据的更新将写入新的页面。随着数据逐渐写入，闪存存储设备中布满了已被更新的无效数据。这部分无效数据所占用的物理地址空间将由 FTL 通过垃圾回收的操作进行回收。

(3)磨损均衡：闪存的擦除次数有限，频繁使用同一个块将会导致该块寿命缩短，闪存存储设备随之损失空间容量。为避免该情况，FTL 通过磨损均衡使所有块的擦除次数相近。磨损均衡分为动态磨损均衡和静态磨损均衡，前者是在选择写入数据的块时，优先选择擦除次数低的块；后者是对长期未擦除的块进行垃圾回收，使寿命长的块可被使用。

随着技术的发展，闪存存储设备从每个存储单元存储 1 位数据(Single-Level Cell，SLC)提升到存储 4 位数据(Quad-Level Cell，QLC)。另外，闪存结构也从二维平面闪存发展至三维堆叠式闪存。这些技术的发展使闪存的密度增加、容量增加和价格下降的同时，也导致闪存的性能下降、可靠性变差且寿命缩短。为解决这一矛盾，混合式闪存应运而生。混合式闪存通过同时存在 SLC 和高密度存储单元，如 QLC(本文中均以 QLC 代表高密度存储单元)，使闪存存储设备同时具有大容量、低价格、高性能、高可靠性和长寿命的特性。因此，混合式闪存是当今消费级终端领域的主流存储设备，如 Intel 665P、Micron Crucial P1、Intel 670P 和 Intel Optane H10 均使用了混合式闪存。研究混合式闪存是进一步提升消费级终端的用户体验的关键。但混合式闪存中存在两种具有不同特性的存储，导致 FTL 算法逻辑中需要增加数据放置、介质间数据迁移等操作，设计更为复杂和困难。

本文针对市场目前最新的混合式闪存设备，首先介绍了混合式闪存的架构和特性，通过理论分析发展方向；其次通过真实设备的实验数据来

挖掘其中存在的问题；最后，介绍国内外的相关研究进展，并进行比较分析，以期为学术界和产业界对混合式闪存的进一步研究提供一定参考价值。

2　混合式闪存的架构

混合式闪存中存在两种架构，一种是 SLC 与 QLC 完全独立，采用不同的介质，如 Intel Optane H10 中 SLC 采用与 QLC 不同的 3D Xpoint 介质；另一种是采用模拟 SLC 技术（pseudo SLC，pSLC），即通过 QLC 只存储 1 位数据来模拟 SLC，其中 SLC 与 QLC 共享通道、芯片、晶元、平面和块。相较而言，模拟 SLC 技术具有更高的性价比，在消费级终端领域更受欢迎，如 Intel 665P、Micron Crucial P1 和 Intel 670P 均使用模拟 SLC 技术。因此，本文以模拟 SLC 技术为混合式闪存架构，其中本节将介绍模拟 SLC 技术、数据通路和该架构所具有的特点。

2.1　模拟 SLC

相较于 SLC，QLC 每个存储单元所存的数据位数更多，导致其阈值电压更密集，难以区分。同时，电荷之间的相互干扰也会更大，因而 QLC 的访问性能、寿命和可靠性均远逊色于 SLC。相关研究表明，SLC 与 QLC 的读写延迟的差距在 20 倍左右，擦除寿命在 30 倍左右[1-2]。为了弥补两者之间的差距，提出了模拟 SLC 技术。

模拟 SLC 技术通过仅存储 1 位数据到 QLC，减少了阈值电压分布，同时降低了电荷之间的干扰，使其具备接近 SLC 的访问性能、寿命和可靠性。与独立 SLC 和 QLC 组成方式不同的是，模拟 SLC 可以与 QLC 随时相互转换，以满足不同场景下的需求。图 1 展示了基于模拟 SLC 技术的混合式闪存架构。在每个平面中具有多个块，部分块中的 QLC 采用模拟 SLC 模式，其余采用 QLC 模式。模拟 SLC 分为两部分，一部分是静态 SLC，其中的模拟 SLC 不会转换成 QLC；另一部分是动态 SLC，其中的模拟 SLC 随着混合式闪存中存储数据量的增大而动态转换成 QLC，也会随着存储数据量的减少转换回 SLC。以 1 TB 的 Intel 665P（初始包含 12 GB 静态 SLC 和 120 GB 动态 SLC，其余为 QLC）为例，所有模拟 SLC 均匀分布在各个平面中。当混合式闪存内数据量多时，120 GB 动态 SLC 将转换成 QLC 来增加可存储容量；当数据量少时，QLC 可转换回动态 SLC 来提升混合式闪存

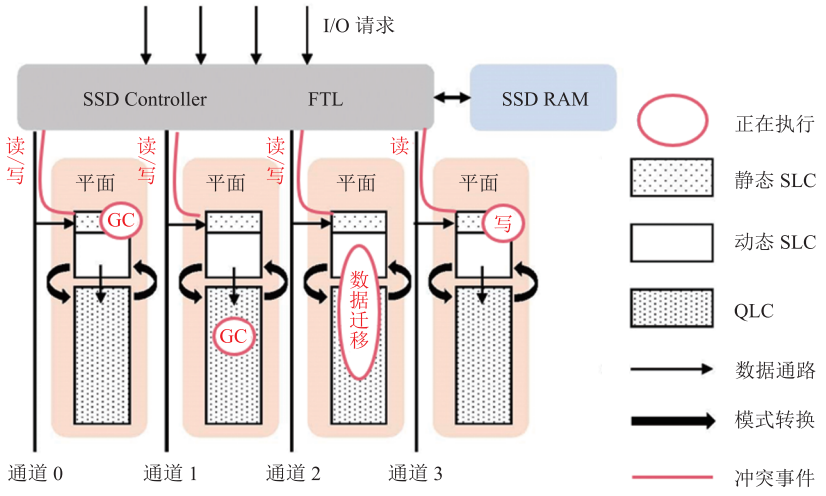


图 1　基于模拟 SLC 技术的混合式闪存架构

Fig. 1　The architecture of hybrid flash memory based on pseudo SLC

的性能，这个过程最多可转换至 120 GB 动态 SLC。因此，通过同时使用模拟 SLC 和 QLC，混合式闪存设备可同时具备高性能、大容量和高可靠性的特性。

然而，设备空间占有率的提升会使混合式闪存设备的性能和可靠性下降，用户在使用过程中会逐渐感受到设备卡顿等情况，影响体验感。为缓解该问题，应对混合式闪存设备中的数据进行精细的组织管理，避免无效数据占用设备空间，并充分利用介质特性。

2.2 数据通路

2.2.1 写数据通路

由于模拟 SLC 具有更好的性能和可靠性，现有的混合式闪存设备通常使用模拟 SLC 作为前端缓冲区，即所有的数据先写入模拟 SLC 来提升性能，同时利用高擦除次数的特性来保护 QLC 不被快速地磨损消耗。QLC 作为后端主存，存储模拟 SLC 中剔除的数据。将数据从模拟 SLC 中剔除到 QLC 的操作称为数据迁移。触发数据迁移的条件通常有两个：设备空闲时和 SLC 容量空间不足时。写数据通路如图 2 中①所示，主机下发的写请求先写入模拟 SLC 中，QLC 中的数据写入是由模拟 SLC 中的数据迁移所带来的。

当 SLC 容量空间不足时，由于闪存擦除是以块为粒度，因此数据迁移将至少以块为粒度，将其中的有效数据迁移至 QLC 后擦除该块。因此，数据迁移的开销相较于数据访问是巨大的。然而，模拟 SLC 中的空间有限，当数据逐步写入时，将会产生数据迁移与主机数据写入的冲突。这将对主机数据写入产生巨大的影响，降低设备性能。如图 1 所示，通道 0 和通道 1 分别表示当 SLC 和 QLC 中发生垃圾回收(Garbage Collection，GC)时与普通访问的读写请求发生冲突，通道 2 表示当 SLC 与 QLC 之间发生数据迁移时与普通访问的读写请求发生冲突。因此，对于混合式存储设备，数据的写入策略应根据数据的属性和设备的使用情况进行精细设计，避免内部数据迁移对设备性能的影响。

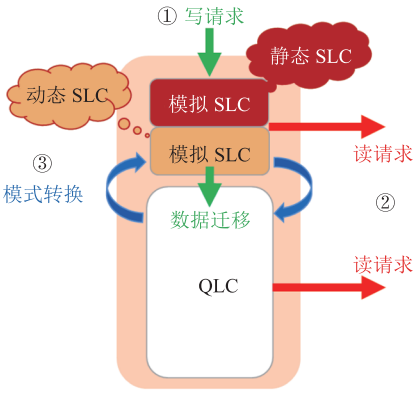


图 2 混合式闪存的数据通路

Fig. 2 The data path of hybrid flash memory

2.2.2 读数据通路

与写数据通路不同的是，读取数据的通路取决于数据存放的位置。在设备的使用过程中，部分数据由模拟 SLC 迁移至 QLC。因此，从混合式闪存设备中读取数据时，可能会从模拟 SLC 和 QLC 中进行读取，如图 2 中②所示。闪存在读取数据时会向相邻的页施加通路电压，因此当一个页面数据被频繁读取时，会导致其他页面的数据出现错误读取的情况。同时，由于 QLC 中的电压分布状态更多、状态之间的电压间隔更小，因此 QLC 中的读干扰现象相较于模拟 SLC 更为严重。

传统解决 QLC 中读干扰严重的方式有两种。第一种是记录页面读取次数，将即将产生干扰的数据重新写入，但是这种方式会产生额外的写入操作，影响寿命。第二种是降低通路电压，降低读取数据时对其他页面的影响，但是过低的通路电压可能导致读取失败。在混合式闪存设备中，模拟 SLC 为缓解读干扰提供了新的思路。

2.2.3 读写冲突

闪存由于读取延迟和写入延迟差异较大，当写请求阻塞读请求时，会导致读请求的延迟大幅增加，影响用户体验，这种情况称为读写冲突，

如图 1 通道 3 所示。在混合式闪存设备中，模拟 SLC 和 QLC 共享通道、芯片、晶元和平面，容易导致模拟 SLC 和 QLC 的请求之间产生冲突。由于模拟 SLC 和 QLC 的延迟差异很大，在混合式闪存设备中读写冲突的问题更加严重。

在传统的解决方案中，读优先策略被广泛采用。当读请求与写请求同时在队列中等待时，读请求将被优先处理以避免被写请求阻塞。而在混合式闪存中，除了读写请求属性外，模拟 SLC 与 QLC 的性能差异使调度成为一个更复杂的问题。其余在单种存储介质下的读写冲突解决方案在混合式闪存中也变得更为复杂，需要考虑的因素更多。因此，针对混合式闪存读写冲突问题的解决方案需要进一步探索。

3　关键问题测试与分析

为了进一步探索混合式闪存的特性和存在的问题，本文设计了多个实验，涵盖不同层级，包括文件系统和块设备层，以及不同场景(标准负载、真实负载、应用启动和开启内存交换)。从写特征、读特征、读写冲突和容量特征 4 个角度对混合式闪存设备中存在的问题进行分析。

3.1　实验配置

本文使用 Intel 665P、Intel 670P 和 Micron Crucial P1 作为混合式闪存，使用某厂商企业级固态硬盘 SSD-A[1]作为 SLC 的纯盘，即仅包含 SLC 一种介质，使用某厂商企业级固态硬盘 SSD-B[1]作为一个存储单元存储 3 位数据(Triple-Level Cell，TLC)的纯盘。Fio[3]用于生成块设备层各种类型的读写负载。IOzone[4]用于生成各种基本文件操作并测试性能。Filebench[5]用于模拟重放应用程序产生的文件系统层负载。Blktrace[6]用于统计每一条请求的延迟信息。Debugfs[7]用于确定每一个逻辑地址对应的具体文件。同时，本文在多个文件系统下进行测试。其中包括第四代扩展文件系统(Fourth Extended Filesystem，EXT4)、闪存友好文件系统(Flash-Friendly File System，F2FS)和 B 树文件系统(B-Tree File System，BTRFS)。

3.2　写特征

3.2.1　文件系统层

为了从文件系统层挖掘混合式闪存的写特征，本文通过 IOzone 默认顺序写和随机写模式向混合式闪存和两种纯盘下发大小为 8 G 的文件系统级标准写负载，并设置混合式闪存的设备占用率为 10% 和 90%。实验结果如图 3 所示，其中纵坐标是以 10% 填充率的混合盘带宽为基准，不同固态盘带宽相较于其带宽的比例。在多种文件系统下，混合式闪存在低设备占用率下写

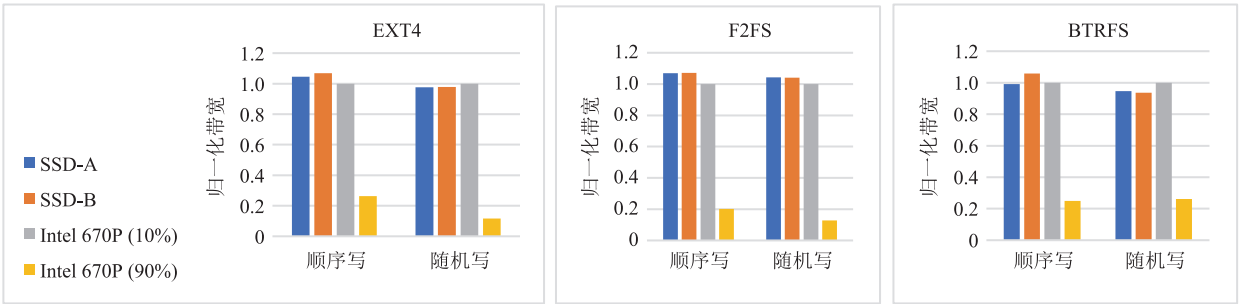


图 3　文件系统层写特征实验结果

Fig. 3　The experimental results of write characteristics in filesystem layer

注[1]：基于厂商信息保密，此处略去厂商信息

性能与纯盘相近；但在高设备占用率下，写性能明显弱于两种纯盘。这是因为在高设备占用率下，模拟 SLC 中的空间已被占满，当继续写入数据时会触发内部的数据迁移，影响主机端的写性能。对于 SSD-A 和 SSD-B 两种介质的纯盘，其性能差异不大，这是因为 TLC 页面读取速度虽然较慢，但是 TLC 访问粒度大，使其具有更高的并行性来获得性能提升，弥补与 SLC 之间的性能差异。

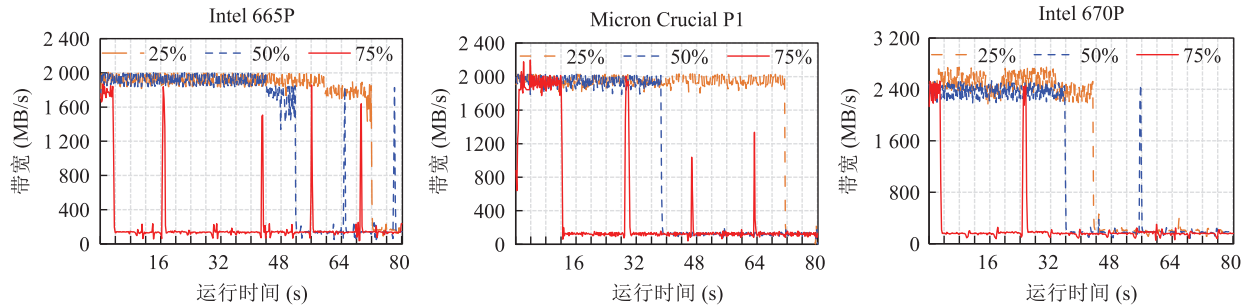
3.2.2 块设备层

为了从块设备层挖掘混合式闪存的写特征，本文通过 Fio 向混合式闪存下发块设备级的随机写和顺序写的标准负载，观察混合式闪存的表现。其中，顺序写的队列长度为 64，请求大小为 1 MB；随机写为随机写入指定文件 5 次，请求大小为 4 KB。分别向混合式闪存设备中填充不同设备容量百分比(25%～75%)的数据，图 4 是混合式闪存在顺序写标准负载下的表现。实验结果显示，在多个混合式闪存下，无论设备填充率为多少，当数据持续写入 SLC 时，混合式 SSD 的性能比单独的 SLC 或 QLC 都差。这同样是因为模拟 SLC 的空间有限，当数据持续写入 SLC 时会触发数据迁移，导致性能坍塌。图 5 是混合式闪存在随机写标准负载下的表现。实验结果显示，混合式闪存在随机写负载下会产生性能波动。这是由于模拟 SLC 的容量较小，造成随机写入的数据无法在模拟 SLC 区域中得到充分的更新，从而导致低效垃圾回收，并产生性能波动。

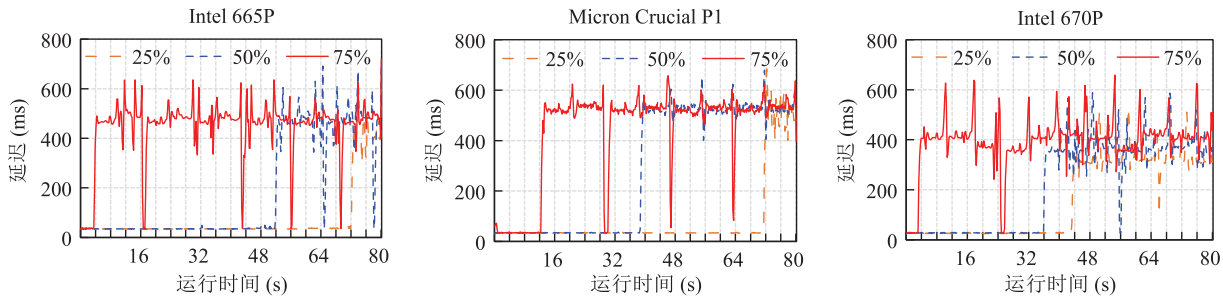
3.3 读特征

3.3.1 文件系统层

本文通过 IOzone 默认顺序读和随机读模式向混合式闪存和两种纯盘下发大小为 6 GB 的文件系统级标准读负载，同时设置设备占用率为 10% 和 90%，来从文件系统层挖掘混合式闪存的读特征。实验结果如图 6 所示，无论是在低设备占用率还是高设备占用率下，混合式闪存的读性能表现稳定。这是因为内部数据迁移等操作是由



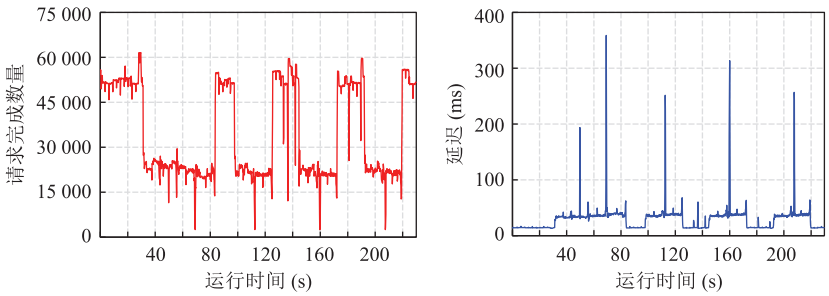
(a) 混合式闪存在持续写入时的带宽



(b) 混合式闪存在持续写入时的延迟

图 4 块设备层顺序写实验结果

Fig. 4 The experimental results of sequential writes in block device layer



(a) 每秒完成的请求数量　　(b) 每秒请求完成的平均延迟

图 5　块设备层随机写实验结果

Fig. 5　The experimental results of random writes in block device layer

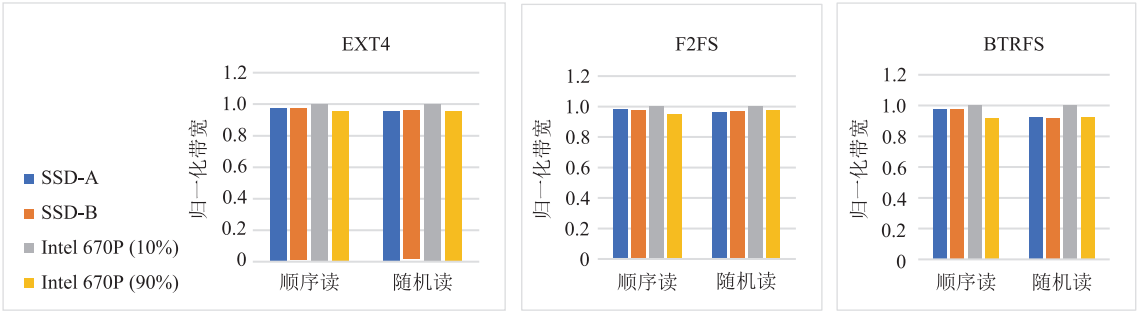


图 6　文件系统层读特征实验结果

Fig. 6　The experimental results of read characteristics in filesystem layer

写入数据所触发的，而在标准读负载下，不会触发混合式闪存中的内部数据迁移等操作。

3.3.2 块设备层

为了从块设备层探寻混合式闪存具有的读特征，本文设计使用 Fio 读取在混合式闪存中保存时间不同的数据，实验结果如图 7 所示。其中，情况 1 为读取刚写入混合式闪存的数据，其延迟相应为模拟 SLC 延迟；情况 2 为读取已写入混合式闪存一段时间的数据，其延迟相应为 QLC 延迟。图 7 中任意坐标点 (x, y) 的含义为所有读请求中 x% 的读请求的完成延迟低于 y。如 (50.00%, 58) 和 (50.00%, 128) 的含义分别为在情况 1 下 50.00% 的读请求延迟低于 58 μs，而在情况 2 下 50.00% 的读请求的完成延迟低于 128 μs。从实验结果可以看出，混合式闪存中读取不同保存时间数据的延迟不同。这是由于当数据在混合式闪存中保存一段时间后将会被迁移至 QLC 中。同时，由于混合式闪存的真实设备会从 QLC 中直接读取数据，并且真实设备的读取策略不会将热数据缓存至模拟 SLC 中，性能和可靠性将成为问题。具体而言，QLC 相对较差的性能会影响热数据的访问性能，同时相较于 SLC 而言，频繁从 QLC 中读取更容易导致读干扰产生，进而影响设备的可靠性。

3.3.3 应用启动运行

为了探索应用启动运行时的读特征，本文首先对混合式闪存进行日常使用，之后对应用启动的关键数据进行读取，实验结果如图 8 所示。从实验结果可以看出，关键数据 (Libmerge.so) 的读取延迟为 QLC 的读取延迟。在日常使用中，混合式闪存启动相关的关键数据会被迁移到 QLC 区域，且该过程用户不可知。这将导致用户在日常使用过程中发现，刚安装的应用启动运行速度快，而使用一段时间的应用启动运行速度慢。

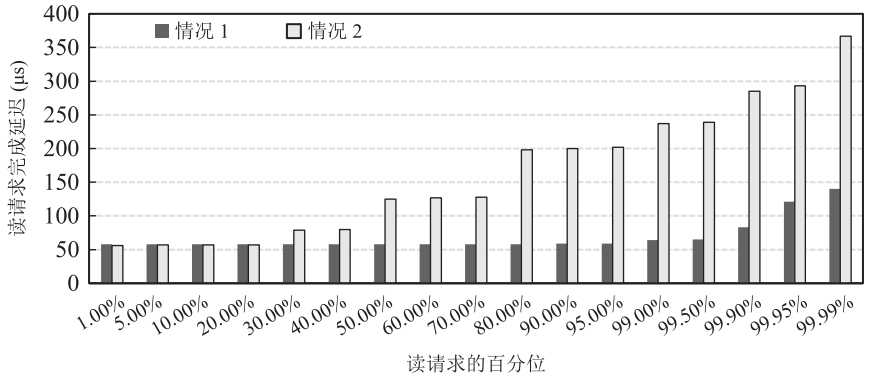


图 7 块设备层读特征实验结果

Fig. 7 The experimental results of read characteristics in block device layer

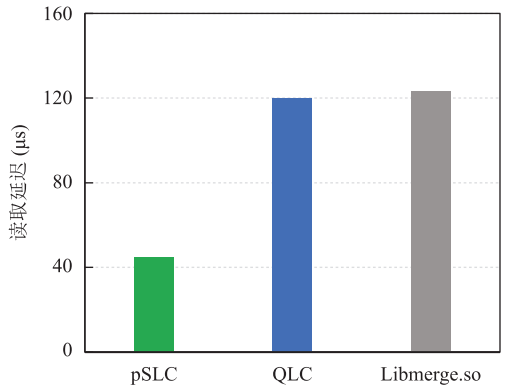


图 8 关键数据访问延迟

Fig. 8 The access latency of critical data

3.4 读写冲突

3.4.1 文件系统层

本文通过 Filebench 对混合式闪存和 SLC 纯盘下发不同读写比例的文件系统级真实负载，观察混合式闪存的读写混合性能表现。真实负载的特征如表 1 所示，实验结果如图 9 所示。其中，实线表示纯盘在各文件系统下的性能，虚线表示混合盘在各文件系统下的性能。从实验结果可以看出，对于非读密集型读写混合负载，纯盘的性能明显优于混合式闪存。这是由于混合式闪存中多种介质的访问延迟不同，使其读写冲突现象更严重。

3.4.2 块设备层

本文通过 Fio 下发不同比例的读写请求，观察混合式闪存的读性能表现，来探索读写冲突的特点，实验结果如图 10 所示。其中，负载大小为 64 MB，采用随机读写模式，队列长度为 64，请求大小为 4 KB。图 10 将所有读请求按照延迟从小到大顺序排列，显示位于所有读请求百分比位置读请求完成延迟，RW 表示读写请求占比，W90＋ 表示写入数据量将会触发混合式闪存中的数据迁移。从图中可以看出，纯盘仍然存在读写冲突问题，但混合式闪存因性能差异和内部数据迁移的影响，使得读性能进一步恶化。

3.4.3 应用启动运行

为了观察应用启动运行过程中混合式闪存的读写冲突表现，本文设计通过对混合式闪存施加

表 1 真实负载的特征

Table 1 The characteristics of real traces

负载名称	文件数量(个)	文件大小(K)	线程数量(个)	读/写/增/删	文件同步
Fileserver	60 000	128	50	1/1/1/1	否
Varmail	1 000	16	16	2/0/2/1	是
Webproxy	10 000	16	100	5/0/1/1	否

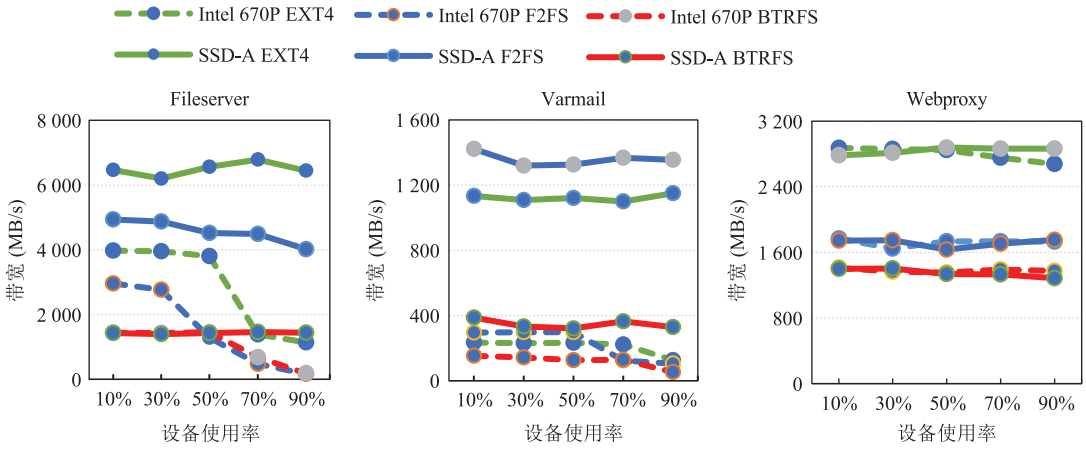


图 9　文件系统层读写冲突实验结果

Fig. 9　The experimental results of read/write interference in filesystem layer

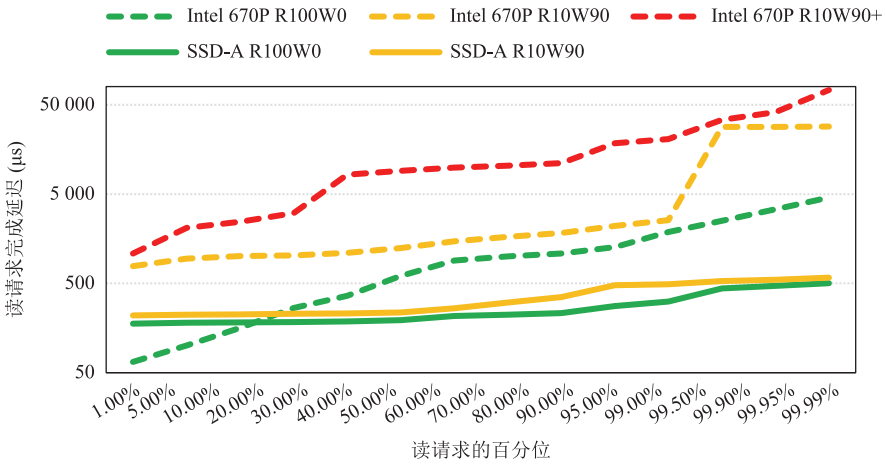


图 10　块设备层读写冲突实验结果

Fig. 10　The experimental results of read/write interference in block device layer

内存压力和 IO 压力，观察读请求的性能表现。对于应用启动，通过提前启动多个应用来触发内存交换(图中的 SWAP)从而施加内存压力，以及通过下载视频(图中的 MD)来施加 IO 压力，实验结果如图 11 所示。从图中可以看出，后台进程的写请求会导致应用启动过程中读请求产生严重尾端延迟问题。如图 12 所示，对于应用运行，通过同时运行多个应用来施加内存压力(图中的 case 2)。从图中可以看出，后台内存交换进程的写请求也会导致应用运行过程中严重尾端延迟问题。

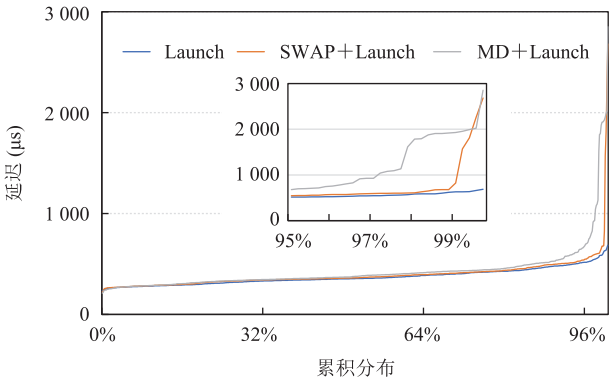


图 11　应用启动请求延迟的累积分布

Fig. 11　Latency cumulative distribution of application launching

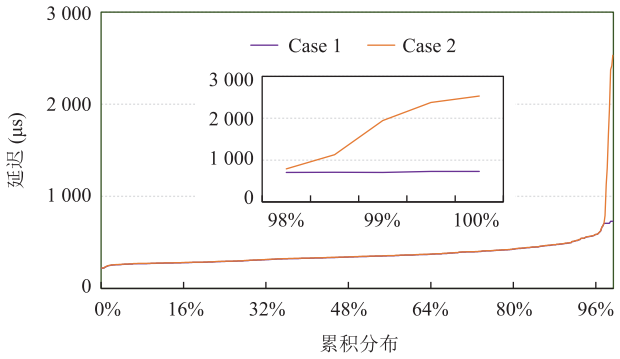


图 12 应用运行请求延迟的累积分布

Fig. 12 Latency cumulative distribution of application running

3.5 容量特征

为了探索混合式闪存的容量特征，即写入数据量对混合式闪存带来的影响，本文通过 Fio 向混合式闪存中缓存不同数据量(占设备容量的百分比)的数据，观察缓存数据的数据量对混合式闪存性能的影响。其中，顺序写请求大小为 1 MB，随机写请求大小为 4 KB，队列长度都设置为 32。实验结果如图 13 所示。对于数据缓存，本地访问性能高于网络带宽使缓存数据的访问性能应随着缓存数据量的增加而增加。然而，从图中可以看出，随着缓存数据量的增加，混合式闪存中缓存数据的访问性能反而下降。这是因为随着缓存数据的增加，模拟 SLC 会更多地转换成 QLC，从而导致混合式闪存的整体性能下降。

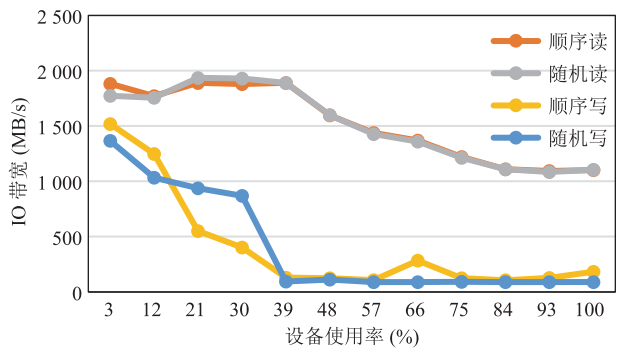


图 13 缓存数据性能表现

Fig. 13 The performance of cached data

3.6 总结

本节对真实混合式存储设备进行多个层次(文件系统层和块设备层)、多种应用场景(各种真实负载和标准负载)和多角度(写特征、读特征、读写冲突和容量特征)的全方位详细分析，挖掘其中的特性和存在的问题。对于写特征，无论从文件系统还是块设备层，混合式存储更易因内部数据迁移导致主机请求访问性能下降。对于读特征，QLC 中的直接读取现象会影响设备的性能和可靠性。对于读写冲突，因混合式设备中各请求延迟差异大以及数据迁移等行为更多，导致读写冲突的现象更明显。对于容量特征，在混合式设备中缓存数据量增多并不能带来更好的性能。

4 前沿研究进展

针对以上混合式闪存中存在的问题，本节将介绍相关的解决思路和最新的研究进展。针对写特征相关研究，主要分为如何充分利用模拟 SLC 加速数据写入的同时避免性能坍塌，以及如何降低由于模拟 SLC 容量较小导致的低效垃圾回收的问题。针对读特征相关研究，主要分为如何充分利用模拟 SLC 加速读性能，以及如何避免 QLC 的频繁读取导致的读干扰问题。针对读写冲突相关研究，主要是如何避免读写数据通路发生争用。针对容量特征相关研究，主要是如何控制混合式闪存的数据落盘从而避免设备占用率高导致的过多模拟 SLC 转换成 QLC。

4.1 写特征

根据混合式闪存架构的分析和实验数据结果可知，模拟 SLC 的容量较小，当写入数据量过多时会导致数据从模拟 SLC 向 QLC 迁移，这时混合式闪存的整体性能将会受到较大影响。为解决该问题，现有的研究方案集中于开辟直接写入 QLC 的数据通路[8-11]。但由于 QLC 的性能、可靠性和寿命等问题，需要对 QLC 中写入的数据进行控制。Stoica 等[8]通过建立理论模型论证了

进行冷热数据分离的必要性，直接将冷数据写入 QLC 中将带来较大的整体性能提升。这是由于冷数据的访问次数较少，若存储在模拟 SLC 中会占用空间，减少对模拟 SLC 的数据访问，同时导致数据迁移的发生。然而，对于冷热数据的识别仅是通过预先标识，并未提供冷热数据识别的方法。Shi 等[9]根据请求大小来对数据进行冷热分类。这是因为元数据多为小请求，且对于性能的影响更为关键。因此当请求下发时，根据当前请求的大小处于历史请求大小中的位置，决定该请求是否为热请求。Li 等[10]综合考虑请求的访问次数和最近访问时间来判断请求的热度，通过一定长度的最近最少使用（Least Recently Used，LRU）链表来记录请求的最近访问时间，同时会记录请求的访问次数。HyFlex[11]中进一步增加设置了触发数据写入 QLC 的条件。其认为数据坍塌往往是模拟 SLC 空间不够用导致的，因此仅当混合式闪存中有效数据量快速增长时才有模拟 SLC 空间不够用的风险，此时才将部分数据写入 QLC。这样可以充分利用模拟 SLC 的性能，同时避免性能坍塌情况的出现。然而，现有的数据识别策略均还未达到准确识别的程度，因此未来如何对数据识别放置才能充分利用好模拟 SLC 和 QLC 的特性仍有广阔的研究空间。

另一方面，由于模拟 SLC 空间区域较小，数据无法得以充分更新，导致很多后续将被更新的数据提前写入 QLC。此时会带来额外的数据写入，加速混合式闪存的磨损，同时模拟 SLC 区域中垃圾回收的效率低将导致混合式闪存的整体性能下降。HyFlex[11]中提出，当模拟 SLC 区域中垃圾回收的效率较低时，利用 QLC 可以动态转换成模拟 SLC 的特性，将部分 QLC 临时转换成模拟 SLC 来提供更多空间进行数据更新，提升垃圾回收的效率。除此之外，该类问题的研究较少，仍需要进行更多的探索来进一步解决该问题。

4.2　读特征

根据读数据通路的理论分析和实验结果可知，混合式闪存中数据会经过数据迁移放入 QLC 中，并可能从 QLC 中被频繁读取，导致严重的读干扰问题。为了解决这一问题，现有的研究方案主要是两种思路，一种是降低通路电压，另一种是进行数据搬移。Cai 等[12]提出根据每个块每天出错的次数和 ECC 可纠错的位数来动态调整页面读取的通路电压，从而最小化页面读取对其他页面造成的读干扰。与之不同的是，Werner 等[13]通过读刷新的方式来解决该问题——当读请求服务时，其相应区域的读计数便会增加，当区域的读计数值达到产生读干扰的限制时，该区域的数据将被写入新的空闲区域，从而避免读干扰带来的影响。Li 等[14]在此基础上优化了数据写入的位置选择，将被频繁读取的数据放入擦除次数少以及累积读取次数少的块。然而，现有关于读干扰的研究仍集中于非混合式闪存，并未考虑如何利用模拟 SLC 的可靠性来更好地解决该问题。

从 QLC 中频繁读取数据除了会带来可靠性问题外，其性能也会降低，尤其是在关键数据被迁移至 QLC 之后，对用户体验的影响更严重。Sun 等[15]提出使用 LRU 链表维护最近访问的数据信息，并通过数据是否位于 LRU 链表来判断数据的冷热，将其中的热数据从 QLC 迁移至快速存储介质。Chen 等[16]提出将小数据文件固定在 SLC 中进行访问。这是因为关键数据往往是小数据，如元数据，同时对于 QLC 而言访问大数据时可以利用其内部的并行性，从而提升性能。因此，将小数据文件固定在 SLC 中可以提升设备的访问性能和用户体验。然而，现有的数据识别方式仍然无法准确判断数据的属性，若过多的数据存储在模拟 SLC 中，将会导致数据迁移，影响混合式闪存的整体访问性能。因此，如何准确识别数据特性从而更好地利用模拟 SLC 和 QLC 的特性，以及如何根据模拟 SLC 的使用状态

来动态调整数据的放置策略将成为此类研究的重难点。

4.3 读写冲突

根据混合式闪存数据通路的理论分析和实验结果可知，由于两种介质的访问性能差异性和架构的复杂性，混合式闪存中的读写冲突相较于纯盘更为严重。为了解决该问题，现有的解决方案主要分为 3 种：基于请求调度、基于数据冗余和基于数据分区。Nguyen 等[17]和 Gao 等[18]通过优先调度读请求来避免写请求长时间阻塞读请求，Wu 等[19]提出当写请求正在执行时读请求被下发——将当前的写请求挂起执行读请求。HotR[20-23]提出将频繁访问的数据复制一份放到专门的区域，当读写请求发生冲突时，从该区域读取数据，从而缓解读写冲突。Lv 等[24]提出大部分数据为只读数据或只写数据，因此将空间划分为读区域和写区域来分别存放相应访问类型的数据，从而避免读写冲突。

现有的这些解决方案并未充分利用混合式闪存的特性。基于请求调度的策略并未将混合式闪存中不同介质读写延迟的不同特性纳入考量，特别是对 QLC 的读取延迟与模拟 SLC 的写入延迟相差不大的情况。基于数据冗余和数据分区的策略并未考虑混合式闪存中具有两种不同介质，因此如何放置不同类别的数据从而更好地利用好不同介质的特性仍然有待探索。

4.4 容量特征

根据混合式闪存架构的理论分析和实验结果可知，当缓存数据量过多时，不仅无法提升缓存数据的访问性能，反而会导致过多的模拟 SLC 转换成 QLC，从而影响混合式闪存的整体性能。为解决该问题，DFCache[25]提出由于缓存数据中大部分网络数据不会被再次访问，因此设计提出缓存文件筛选器和缓存空间管理器。缓存文件筛选器根据数据的访问频次来决定缓存数据是否值得被缓存，缓存空间管理器通过限制缓存空间的大小和设计缓存数据剔除策略来使缓存数据总是小于设备中剩余模拟 SLC 空间的大小，从而避免过多的缓存数据写入混合式闪存。该方法减少了缓存数据量，避免了无用的缓存数据造成混合式闪存的整体性能下降。CacheSifter[26]同样提出很多缓存文件未被使用便被删除，不同的是他提出使用机器学习的方式来对缓存文件进行分类，基于缓存数据的活跃期进行分别管理，只将必要的缓存文件写回混合式闪存，从而提升混合式闪存的性能和寿命。对于混合式闪存容量特征的相关研究，一方面缓存数据的管理仍具有探索空间，考虑如何更准确地设计识别关键缓存数据，从而提升缓存数据的访问性能，同时避免过多缓存数据影响混合式闪存的整体性能；另一方面需要探索更多类型数据进行针对性识别管理，避免容量特征的相关问题。

5 结论与展望

混合式闪存已成为消费级终端的主流存储设备，然而现有研究仍未对其中所存在的问题和特性进行充分的探索。本文从混合式闪存的架构出发，从多个层级（文件系统层和块设备层）、多个场景（各种真实负载和标准负载）以及多个角度（读特征、写特征、读写冲突和容量特征）探索其中存在的真实问题，并针对这些问题的现有研究进行了归纳和梳理。同时，进一步对混合式闪存的写特征、读特征、读写冲突和容量特征的现有研究进行综合比较，探讨和评述未来的发展方向。对于混合式闪存而言，如何对其中的数据进行准确识别划分从而更好地利用两种介质的特性是未来的关键问题。

参 考 文 献

[1] Kouchi T, Kumazaki N, YamaokaM, et al. 13.5 A 128 Gb 1 b/cell 96-word-line-layer 3D flash

memory to improve random read latency with t_{PROG}=75 μs and t_R=4 μs [C] // Proceedings of the IEEE International Solid-State Circuits Conference, 2020: 226-228.

[2] Khakifirooz A, Balasubrahmanyam S, Fastow R, et al. 30.2 A 1 Tb 4 b/cell 144-tier floating-gate 3D-NAND flash memory with 40 Mb/s program throughput and 13.8 Gb/mm 2 bit density [C] // Proceedings of the IEEE International Solid-State Circuits Conference, 2021: 424-426.

[3] Axboe J. Fio Manpage [EB/OL]. (2017-6-24)[2022-4-13]. https://fio.readthedocs.io/en/latest/fio_man.html#cmdoption-arg-write-bw-log.

[4] Norcott W, Capps D. IOzone Filesystem Benchmark [Z/OL]. 2007. https://www.iozone.org.

[5] McDougall R. Filebench Tutorial [Z/OL]. (2005-10-15)[2022-4-13]. http://www.nfsv4bat.org/Documents/nasconf/2005/mcdougall.pdf.

[6] Axboe J, Brunelle A, Scott N. Blktrace [Z/OL]. (2005-8-26)[2022-4-13]. https://git.kernel.dk/cgit/blktrace/.

[7] Greg KH. Debugfs [Z/OL]. (2004-12-13)[2022-4-13]. https://lwn.net/Articles/115405/.

[8] Stoica R, Pletka R, Ioannou N, et al. Understanding the design trade-offs of hybrid flash controllers [C] // Proceedings of the 27th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems, 2019: 152-164.

[9] Shi L, Di YJ, Zhao MY, et al. Exploiting process variation for write performance improvement on NAND flash memory storage systems [J]. IEEE Transactions on Very Large Scale Integration Systems, 2016, 24(1): 334-337.

[10] Li YK, Shen BB, Pan YB, et al. Workload-aware elastic striping with hot data identification for SSD RAID arrays [J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2017, 36(5): 815-828.

[11] Shi L, Luo LF, Lv YN, et al. Understanding and optimizing hybrid SSD with high-density and low-cost flash memory [C] // Proceedings of the 39th IEEE International Conference on Computer Design, 2021: 236-243.

[12] Cai Y, Luo YX, Ghose S, et al. Read disturb errors in MLC NAND flash memory: characterization, mitigation, and recovery [C] // Proceedings of the 45th Annual IEEE/IFIP International Conference on Dependable Systems and Networks, 2015: 438-449.

[13] Werner J, Cohen ET, Canepa TL. Read disturb handling for non-volatile solid state media: U.S. US20140136884A1 [P/OL]. 2014-5-15[2022-4-2]. https://patents.google.com/patent/US20140136884.

[14] Li J, Huang BW, Sha ZB, et al. Mitigating negative impacts of read disturb in SSDs [J]. ACM Transactions on Design Automation of Electronic Systems, 2021, 26(1): 1-24.

[15] Sun C, Okamoto S, Hachiya S, et al. Design guidelines of storage class memory/flash hybrid solid-state drive considering system architecture, algorithm and workload characteristic [J]. IEEE Transactions on Consumer Electronics, 2016, 62(3): 267-274.

[16] Chen H, Lv YN, Li CL, et al. An empirical study of hybrid SSD with Optane and QLC flash [C] // Proceedings of the 38th IEEE International Conference on Computer Design, 2020: 175-178.

[17] Nguyen DT, Zhou G, Xing GL, et al. Reducing smartphone application delay through read/write isolation [C] // Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services, 2015: 287-300.

[18] Gao CM, Shi L, Zhao MY, et al. Exploiting parallelism in I/O scheduling for access conflict minimization in flash-based solid state drives [C] //

Proceedings of the 30th International Conference on Mass Storage Systems and Technologies, 2014: 1-11.

[19] Wu GY, He XB. Reducing SSD read latency via NAND flash program and erase suspension [C] // Proceedings of the 10th USENIX Conference on File and Storage Technologies, 2012: 1-7.

[20] Wu SZ, Zhang WW, Mao B, et al. HotR: alleviating read/write interference with hot read data replication for flash storage [C] // Proceedings of the Design, Automation & Test in Europe Conference & Exhibition, 2019: 1367-1372.

[21] Elyasi N, Arjomand M, Sivasubramaniam A, et al. Content popularity-based selective replication for read redirection in SSDs [C] // Proceedings of the 26th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems, 2018: 1-15.

[22] He BS, Yu JX, Zhou AC. Improving update-intensive workloads on flash disks through exploiting multi-chip parallelism [J]. IEEE Transactions on Parallel and Distributed Systems, 2015, 26(1): 152-162.

[23] Yan S, Li H, Hao M, et al. Tiny-tail flash: near-perfect elimination of garbage collection tail latencies in NAND SSDs [J]. ACM Transactions on Storage, 2017, 13(3): 1-26.

[24] Lv YN, Shi L, Li Q, et al. Access characteristic guided partition for read performance improvement on solid state drives [C] // Proceedings of the 57th ACM/IEEE Design Automation Conference, 2020: 1-6.

[25] Gu B, Luo LF, Lv YN, et al. Dynamic file cache optimization for hybrid SSDs with high-density and low-cost flash memory [C] // Proceedings of the 39th IEEE International Conference on Computer Design, 2021: 170-173.

[26] Liang Y, Pan R, Ren T, et al. CacheSifter: sifting cache files for boosted mobile performance and lifetime [C] // Proceedings of the 20th USENIX Conference on File and Storage Technologies, 2022: 1-12.