基于 NAND Flash 存储器的磨损均衡 DP 算法优化

薛 镭

(中国船舶重工集团公司第七一五研究所 浙江 杭州 310023)

摘 要 双池 DP(Dual Pool)算法对于 NAND Flash 的磨损均衡控制水平较高,但是磨损均衡过程较长,磨损均衡分布不均,第一个磨穿块容易较早出现。为了解决这些不足之处,在吸收双池算法磨损控制思想的基础上,提出一种基于优先搜索树(PST)的磨损均衡思想,在块搜索策略和垃圾回收策略等方面进行优化。实验结果表明,算法继承了磨损均衡控制水平较高的优点,磨损均衡过程相比双池算法降低 70%,系统资源利用率降低 40%,NAND Flash 使用寿命提高了 30%。

关键词 双池算法 磨损均衡 页复制操作 优先搜索树

中图分类号 TP302 **文献标识码** A DOI:10.3969/j.issn.1000-386x.2019.06.051

OPTIMIZATION OF WEAR EQUALIZATION DP ALGORITHM BASED ON NAND FLASH MEMORIZER

Xue Lei

(*715 Research Institute, China Shipbuilding Heavy Industry Group Corporation, Hangzhou 310023, Zhejiang, China*)

Abstract Dual Pool(DP) algorithm has a high level of wear balance control for NAND Flash, but the wear balance process is long, the wear balance distribution is uneven, and the first wear block is easy to appear early. In order to solve these shortcomings, based on the wear control idea of dual pool algorithm, I proposed a wear balance idea based on priority search tree(PST) to optimize in block search strategy and garbage collection strategy. The experimental results indicate that the algorithm could inherit the advantages of high wear balance control level. Compared with the dual-pool algorithm, the wear balance process reduces by 70%, the utilization rate of system resources reduces by 40%, and the service life of NAND Flash increases by 30%.

Keywords Dual pool algorithm Wear equalization Page replication operation Priority search tree(PST)

0 引 言

用 NAND Flash 作为存储媒介,具有体积小、功耗低、速度快等特点,广泛受到青睐,在手机、平板电脑、数码相机等电子产品中均有运用[1]。NAND Flash 不能被复写,写之前需要做擦除操作。NAND Flash 读写操作的基本单位是页。其中包含最新数据的页被称为有效页(新数据被称为有效数据),包含旧数据的页被称为无效页或脏页,脏页经过擦除操作后成为空闲页,才可以重新写入数据。NAND Flash 擦除操作的基本单位是块(如 1 个块包含多个页),块的擦除次数是有限的(根据不同颗粒类型如 TLC/MLC/SLC 等,一般几千次到 100 万次之间),如果出现块的擦除次数达到了上限,NAND Flash 的性能将大幅下降[2]。

经常使用读写的数据称为热数据,很少被读写到的数据称为冷数据,冷数据和热数据将导致存储在 NAND 介质上的数据块的读写擦除计数出现严重不平衡。为了保持 NAND Flash 性能的稳定,必须提出一种方法使每个块的擦除操作尽可能均衡,这种方法就是磨损均衡算法。

Chang[3] 提出的双池算法是目前众多磨损均衡算法中较为主流的算法。相比其他算法而言,该算法主要解决了“冷块”几乎不被处理和磨损控制水平较低

收稿日期:2018-12-16。**薛镭**,工程师,主研领域:嵌入式系统,电子信息。

的问题,达到了预期的磨损均衡效果。本文在双池算法的基础上,保留磨损控制的策略,针对其不足之处提出进一步优化的方案,并通过仿真实验进行验证分析,期望保留较高磨损控制水平,缩短磨损均衡过程,减少页复制次数,使 NAND Flash 的使用寿命进一步提高。

1　双池算法

1.1　算法原理

该算法首先将 NAND Flash 块随机地分配到“冷池”和“热池”中,随着数据的更新,算法将存放冷热数据的 NAND Flash 块分别存入“冷池”和“热池”中。如果“热池”中的某些块的被擦除次数大于某个阈值,这些块的数据将被将换到“冷池”中擦除次数最少的闪存块中。

比如 NAND Flash 一共有 1 024 个块,系统初始化时,给“冷池”分配序号为奇数的块,给“热池”分配序号为偶数的块如图 1 所示。“冷池”NAND Flash 块表 A 和“热池”NAND Flash 块表 B 分别包含 512 条表项,每个表现有 32 个字节,包含:块序号(4 Bytes) + 擦除次数(4 Bytes) + 有效页比例(4 Bytes) + 修改时间(16 Bytes) + 回收权值(4 Bytes),则表 A 和表 B 大小分别为 16 KB,一共 32 KB。

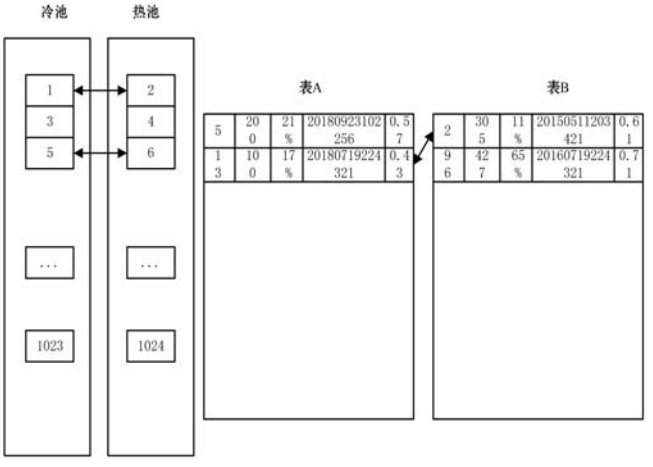

图 1　双池算法磨损均衡

以 MLC 颗粒的 NAND Flash 为例,假设每个块擦除最大次数为 3 000 次,设定降温阈值为 2 000 次。当系统扫描表 B 中第一个(块号 m)擦除次数大于 2 000 的块时,系统扫描表 A 中擦写次数最小的块(块号 n),将块号 m 中的数据与块号 n 中的数据对换。

1.2　块搜索策略

双池算法搜索的块搜索策略采用的是排序算法,根据 DP 算法描述的一次“热块”降温过程,搜索时间包括扫描表 A 查找擦写次数最小的块时间开销 + 扫描表 B 查找第一个擦写次数阈值的块开销。

扫描表 A 或表 B 在空间复杂度最低并且排序稳定的前提下,查找或插入表 A 和表 B 中一个节点的最坏时间复杂度为 $O(N)$,每次查找操作时间为 10 μs。按照最大时间计算:DPT_1 = 0.01 ms × 512 × 2 = 10.24 ms。

1.3　垃圾回收策略

双池算法选用 CAT[4] 算法(Cost-Age-time Algorithm)作为垃圾回收策略。CAT 算法的权重计算公式如下,选择权重最小的块进行回收。

$$\frac{u}{1-u}\times\frac{1}{age}\times CT \tag{1}$$

式中:age 表示此块最后一次修改的时间;u 表示块有效页比例;CT 表示块的擦除次数。

这种策略优点是综合考虑了块擦除次数、有效页比例、块最后一次修改时间等 3 个因素,垃圾回收效果较好;缺点是只考虑物理块最后一次修改时间,不能真实反映物理块年龄,同时对块表的更新操作及页的复制操作频繁,系统开销大。

双池算法占用的空间资源包括块信息表占用资源 + 块搜索占用资源 + 垃圾回收策略占用资源。其中块信息表占用资源包含所有块完整的信息,块搜索占用资源包含所有块的擦除次数信息,垃圾回收策略占用资源包含所有块的权值信息。按照最大空间复杂度计算:

DPS_1 = 32 Byte × 1 024 + 4 Byte × 1 024 × 2 = 40 KB。

2　优先搜索树算法

2.1　算法原理

在优先搜索树(简称 PST 算法)中,每个树的节点由基索引和堆索引来标识。优先搜索树是一个依赖于基索引的搜索树,并附加一个类堆属性即一个节点的堆索引不会小于其子节点的堆索引,任意一个节点的左子节点基索引也都不大于右子节点基索引。

块节点(16 Bytes) = 块序号(基索引 4 Bytes) + 擦除次数(堆索引 4 Bytes) + 有效页比例(附加信息 4 Bytes) + 回收权值(附加信息 4 Bytes)。

比如 NAND Flash 一共有 1 024 个块,则整个索引表的大小为 32 KB。系统初始化时,搜索算法从块序号为 0 的块按照块序号(即基索引)开始搜索,这个搜索树是个二叉树却是不对称的,如图 2 所示。

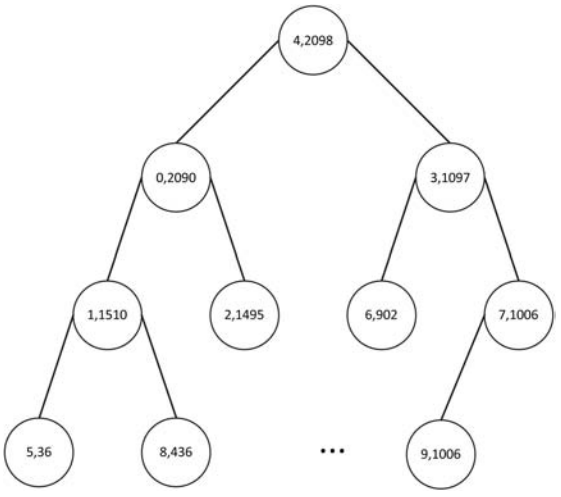

图 2 优先搜索树算法

初始化完成后,所有节点扫描完成后形成二叉树 Cm 树。因此,根据这个二叉树磨均衡操作的过程就是不断从非对称向对称结构进行调整的过程。

2.2 块搜索策略

还是以 MLC 颗粒的 NAND Flash 为例,假设每个块擦除最大次数为 3 000 次,设定降温阈值为 2 000 次。当系统从 Cm 树块序号为 0 的块开始查找第一个擦除次数大于 2 000 的 NAND Flash 块(块号 p),然后从 Cm 树块序号为 0 的块开始查找擦除次数最小的块(块号 q),将块号 p 中的数据与块号 q 中的数据对换,并根据擦除次数调整 Cm 树中节点 p 和 q 的位置。降温一个"热池"NAND Flash 块数据开销:

$PSTT_1$ = 查找块 p 的时间 + 查找块 q 的时间

Cm 二叉树查找或插入一个节点最坏时间复杂度为 $O(\log_2 N)$,每次查找操作时间为 10 μs。

按照最大时间计算:

$$PSTT_1 = 0.01\ \text{ms} \times \log_2 1\ 024 \times 2 = 0.2\ \text{ms}$$

2.3 垃圾回收策略

基于优先搜索树的垃圾回收策略不再考虑每个块的最后一次修改时间,而是将当前块的擦除次数与最大擦除次数的差距作为考虑因素,权重计算如下:

$$\frac{u}{1-u} \times \frac{CT}{CT_{\max} + 1} \tag{2}$$

式中:u 表示有效页比例,CT 表示当前块擦除次数,$CT_{\max}$ 表示最大擦除次数。

这种策略减少了一个权重因子,公平地考虑擦除次数与有效页比例对于磨损均衡的影响,使得回收操作不受块更新时间因素的干扰,同时也减少了权重计算开销。

优先搜索树算法占用的空间资源包括块信息表占用资源 + 块搜索占用资源 + 垃圾回收策略占用资源。其中块信息表占用资源包含所有块完整的信息,块搜索占用资源包含所有块的擦除次数信息,垃圾回收策略占用资源包含所有块的权值信息。按照最大空间复杂度计算:

$$PSTS_1 = 16\ \text{Byte} \times 1\ 024 + 4\ \text{Byte} \times 1\ 024 \times 2 = 24\ \text{KB}。$$

2.4 算法比较

相对于 DP 算法的耗费的系统资源、时间开销也大为降低,如表 1 所示。

表 1 算法比较

比较项目	DP 算法	PST 算法
磨损控制水平	好	好
垃圾回收效率	较好	好
系统响应	慢	快
系统开销	大	小
优点	磨损均衡度高	系统响应快
缺点	系统响应慢,开销大	页操作次数有所降低,但仍然较大

3 实验分析

本文使用 FlashSim[5] 实验平台对设计的算法进行性能分析。

3.1 实验环境

FlashSim 是一款开源的、事件驱动的、模块化的基于 C ++ 的 SSD 模拟器,内置了多种 FTL 策略,能够提供响应时间、能耗的模拟及许多额外的统计信息。

FlashSim 分为两部分,软件部分和硬件部分。硬件部分模拟 SSD 固态盘,其内部结构与 SSD 的实际结构有着很好的映射关系。FlashSim 模拟器硬件部分由处理器、RAM、总线和 flash 芯片组成如图 3 所示。

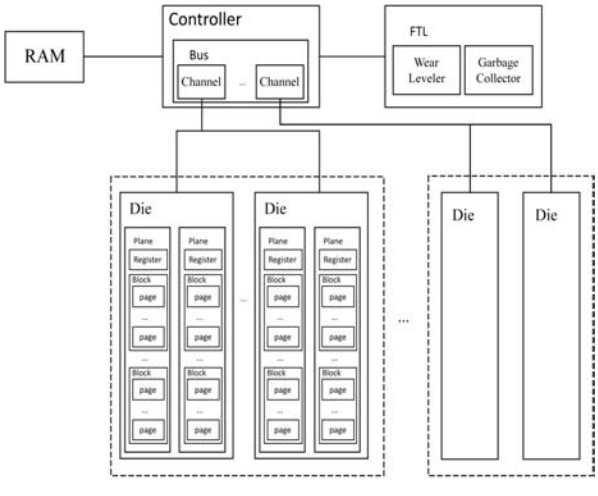

图 3 FlashSim 模拟器硬件结构

设置 FlashSim 模拟器硬件部分 Nand flash 芯片参数如表 2 所示。

表 2　芯片参数

容量	块的个数	块的页个数	页大小	页读延迟	页写延迟	块擦除延迟
512 MB	1 024	256	2 KB	20 ns	200 ns	2 ms

FlashSim 模拟器软件部分模拟 FTL 功能,运行在调试电脑 Ubuntu 10.04 的 Linux 系统上,整个实验平台如图 4 所示。

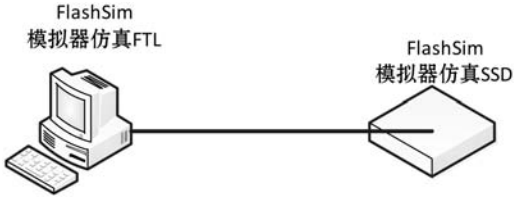

图 4　实验平台

设置 FlashSim 模拟器软件部分实验参数如表 3 所示。

表 3　实验参数

静态数据大小	负载文件大小	并发文件数	事务数	阈值	P_{limit}	P_{wl}
19.7 MB	10 ~ 200 KB	600	2.5 ~ 15 万	2 000	0.5	0.5

3.2　实验数据

为了模拟出真实环境下的算法性能,输入文件的选择主要来源于企业级宽频谱的工作负载[6]。我们采用了两种测试文件,分别是 Financial、Web search。这两种负载文件特性如表 4 所示。

表 4　实验数据

工作负载特效	平均操作/KB	读操作	连续读	连续写
Financial	3.28	32.28%	0.46	1.84
Web Search	43.68	94%	41.2	0.87

对于 Financial 文件数据,是以写为主,具有很强的时间局限性,对于 Web search,其主要是读操作的踪迹数。通过下载的测试源数据包,其中有两份 Financial 测试数据(分别称为 F1、F2)和三份 Web Search 测试数据(分别称之为 W1、W2、W3),对于这五份数据都作为实验的测试数据源。

3.3　性能分析

在实验中,以系统响应时间、块磨损度作为性能指标,以此验证 PST 算法和 DP 算法的性能。

(1) 系统平均响应时间　图 5 是 PST 算法的 FTL 和 DP 算法的 FTL 在不同负载下平均响应时间对比图。由于 PST 算法在系统命中率提高了很多,以至于减少了很多地址转换过程,因此其在系统平均响应时间性能上得到优化如表 5 所示。

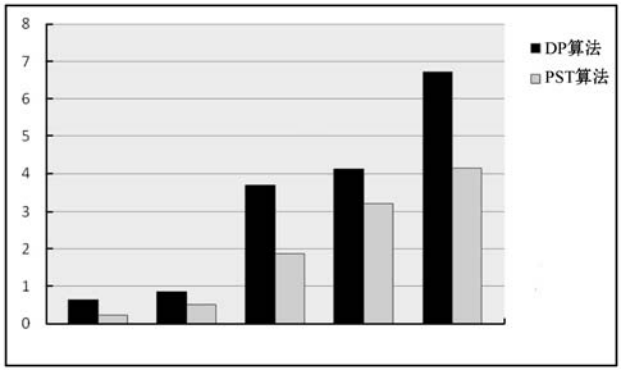

图 5　系统平均响应时间

表 5　系统平均响应时间

数据类型	DP 算法	PST 算法
F1	0.632 581 2	0.236 632 3
F2	0.863 354 7	0.521 197 1
W1	3.689 951 3	1.862 214 2
W2	4.131 258 4	3.223 975 1
W3	6.712 293 1	4.135 589 3

(2) 块磨损度　两种算法分别用以写入为主的 Financial 测试数据(分别称为 F1、F2)运行 5 万次后 Nand Flash 的磨损情况如图 6 所示,横坐标为 Nand Flash 块号,纵坐标为 Nand Flash 块擦除次数。

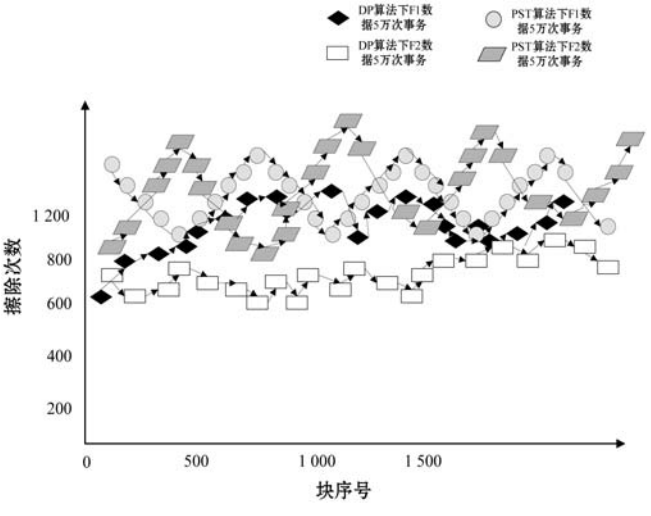

图 6　Nand Flash 的磨损情况

4　结　语

本文提出了一种基于优先搜索树的磨损均衡算法,通过基索引和堆索引的组合进行类似二叉树查找

和排序,针对 DP 算法中磨损过程缓慢和磨损均衡分布不均的问题进行了改进,降低了系统响应时间和提升了块的使用寿命,进一步优化了 FTL 的性能。

参 考 文 献

[1] Douglis F, Caceres R, Kaashoek M F. Storage alternative for mobile computers[C]//Proceeding of the 1st Symposium on operating Systems Design and Implementation(OSDI). 1994: 25 - 37.

[2] Leventhal A. Flash Storage Memory[J]. Communication of the ACM,2008,51(7):47 - 51.

[3] Chang L P. On efficient wear leveling for large-scale flash-memory storage systems[C]//Proceeding of ACM Symposium on Applied Computing. ACM Press,2007: 1126 - 1130.

[4] Chang M, Chang R. Cleaning policies in mobile computers using flash memory[J]. Journal of Systems and Software, 1999,48(3):213 - 231.

[5] http://csl.cse.psu.edu/?q=node/321.

[6] Choi H P, Kim Y S. An Efficient Cache Management Scheme of Flash Translation Layer for Large Size Flash Memory Drives[J]. Journal of The Korea Society of Computer and Information, 2015,20(11):31 - 38.

[7] 刘洋,陆冠群,陈章龙,等. OEBS:一种闪存磨损均衡算法[J]. 小型微型计算机系统,2009,30(12):2489 - 2492.

[8] Nguyen D. NAND Flash Consumption in Tablets to Rise Nearly 400 Percent in 2011[OL]. 2011. http://www.isuppli.com/Memory-and-Storage.

[9] Kwon O,Koh K,Lee J. FeGC:An efficient garbage collection scheme for flash memory based storage systems[J]. Journal of Systems and Software,2011,84(9):1507 - 1523.

[10] Lin M, Chen S. Efficient and intelligent garbage collection policy for NAND flash based consumer electronics[J]. IEEE Transactions on Consumer Electronics, 2013, 59(3): 538 - 543.

(上接第 170 页)

4 结 语

研究表明,利用深度学习技术能够以较高的准确率快速精准地分析 IPTV 用户点播视频类型活跃度。尽管本文的研究已经取得了一定阶段性成果,但是由于模型还不够完善,并且对用户行为分析不够全面。下一步工作将会对用户行为进行全方位分析,包括用户点播视频消费指数、用户观看视频体验质量等,并对以上信息进行行为建模、深度学习和知识发现,具体分析 IPTV 用户的观看视频行为习惯和兴趣爱好,从整体上洞悉用户的需求、强化客户关怀,使得运营商可进一步为用户提供更有价值的、个性化的服务。

参 考 文 献

[1] Degrande N, Laevens K, Vleeschauwer D D, et al. Increasing the User Perceived Quality for IPTV Services[J]. IEEE Communications Magazine, 2008, 46(2): 94 - 100.

[2] 顾军华,高星,王守彬,等. 基于大数据的 IPTV 视频评估模型[J]. 计算机应用与软件,2018,35(8): 231 - 237.

[3] Kerpez K, Waring D, Lapiotis G, et al. IPTV service assurance[J]. Communications Magazine IEEE, 2006, 44(9): 166 - 172.

[4] Acquisti A, Brandimarte L, Loewenstein G. Privacy and human behavior in the age of information[J]. Science, 2015, 347(6221): 509 - 514.

[5] 汪敏娟, 吕超. 基于点播行为特征指数的 IPTV 用户精准分群算法[J]. 电信技术, 2017(7): 24 - 26.

[6] 唐明. 基于用户偏好的视频推荐技术研究[D]. 成都:电子科技大学, 2013.

[7] Hei X, Liang C, Liang J, et al. Insights into PPLive: A Measurement Study of a Large-Scale P2P IPTV System[J]. IEEE Transactions on Multimedia,2007,9(8):1672 - 1687.

[8] Wu X, Zhu X, Wu G Q, et al. Data mining with big data [J]. IEEE Transactions on Knowledge & Data Engineering, 2013, 26(1): 97 - 107.

[9] Tang S, Lu Y, Hernández J M, et al. Topology Dynamics in a P2PTV Network[M]. Springer Berlin Heidelberg, 2009: 326 - 337.

[10] He K, Zhang X, Ren S, et al. Deep Residual Learning for Image Recognition[C]//2016 IEEE Conference on Computer Vision and Pattern Recognition(CVPR). 2016:770 - 778.

[11] 罗海艳, 杨勇, 王珏, 等. 基于人工蜂群改进的 BP 神经网络移动用户行为分析及预测方法[J]. 沈阳农业大学学报, 2015(6): 757 - 761.

[12] 汤荣志, 段会川, 孙海涛. SVM 训练数据归一化研究[J]. 山东师范大学学报(自然科学版), 2016, 31(4): 60 - 65.

[13] Senior A, Lopez-Moreno I. Improving DNN speaker independence with I-vector inputs[C]//2014 IEEE International Conference on Acoustics, Speech and Signal Processing(ICASSP). IEEE, 2014: 225 - 229.

[14] Pinheiro P H O, Collobert R. Recurrent Convolutional Neural Networks for Scene Parsing[EB]. arXiv: 1306.2795, 2013.