Future Generation Computer Systems 90 (2019) 461–476

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

CalmWPC: A buffer management to calm down write performance cliff for NAND flash-based storage systems

Hui Sun [a,*], Guodong Chen [a], Jianzhong Huang [b], Xiao Qin [c], Weisong Shi [d]

[a] School of Computer Science and Technology, Anhui University, Hefei 230601, China
[b] School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074, China
[c] Department of Computer Science and Software Engineering, Auburn University, Auburn, AL 36834, USA
[d] Department of Computer Science, Wayne State University, Detroit, MI 48202, USA

ARTICLE INFO

Article history:
Received 5 April 2018
Received in revised form 24 June 2018
Accepted 7 August 2018
Available online xxxx

Keywords:
NAND flash
Solid state disk
Buffer management
Write performance cliff
Prediction
Fingerprint database

ABSTRACT

NAND Flash-based solid state disks (*i.e.*, SSDs) are widely applied in large-scale storage systems. However, NAND Flash is featured with the asymmetric read and write performance, high erase latency, and the limited number of program/erase cycles (P/Es). Under random write-intensive workloads, a garbage collection (*i.e.*, GC) process inside SSDs causes write performance cliff, which causes high latency for I/O access and degrades SSD lifetime. In real-time transactional applications, such large write performance cliff affects the response time of I/O requests, thereby leading to serious critical errors in real-time applications.

To handle this issue, we propose a buffer management strategy called CalmWPC to *calm* down SSD *w*rite *p*erformance *c*liff. CalmWPC seamlessly integrates a data cluster-based data management, a historical access-based prediction algorithm, a semantic fingerprint database. The prediction algorithm checks the future data-cluster activity while classifying the cluster based on its historical write operations. The fingerprint database stores semantic messages for write/update between the buffer and NAND Flash memory. With the fingerprint database in place, CalmWPC calculates the number of invalid data pages in a block in real time. CalmWPC flushes the data cluster into flash memory when the number of update pages reaches a predefined threshold. Our CalmWPC optimizes write performance cliff during GC under random-write workloads.

Experimental results reveal that CalmWPC is able to reduce write performance cliff, improve the average latency of user I/Os, and optimize write amplification. Take Financial1 as an example, CalmWPC reduces the write performance cliff by averages of 60.9% and 60.0% compared with LRU and CFLRU. CalmWPC also shortens the response time of LRU and CFLRU by averages of 69.4% and 70.1%, respectively.

1. Introduction

In SSD-based storage systems, write performance cliff [1,2] happens to the point where the write rate is higher than that a storage device's write-cache can sustain, translating to a steep increase in latency. This phenomenon gets worse when an application contains more random-write I/Os. In this manuscript, we design a novel buffer management strategy to decrease write performance cliff and improve SSD performance.

With the increasing applications of big data, HDD-based storage systems are unable to meet the requirements of big data applications for low response time. A HDD includes mechanical components that require a long time to address. However, the electronic medium-based SSDs have low mapping-address overload because NAND Flash has higher performance and less power consumption than HDDs [3–5]. With the popularity of 3D large-capacity NAND Flash [6], SSDs are increasingly employed in various high-performance and large-scale storage systems to process big-data applications [7,8].

However, erase-before-write and out-place-update operations [9] in NAND Flash degrade SSD performance and lifetime. This issue posed by the inherent characteristics in NAND Flash has become increasingly apparent to the widespread use of NAND Flash-based SSDs. The update data writes into an erased data block, and the corresponding original data page is set to be invalid in NAND Flash. When the user space reduces to a threshold, invalid data space will be reclaimed. This process is called garbage collection (GC for short) which includes copy-rewrite and block-erase operations. Moreover, the copy-rewrite operation may conflict with the external user I/Os for data buses [10], which causes write performance cliff and greatly increases user response time. The erase time in flash memory is much longer than that in user response time [11].

* Corresponding author.
E-mail addresses: sunhui@ahu.edu.cn (H. Sun), gdchen_ahu@126.com (G. Chen), hjzh@hust.edu.cn (J. Huang), xqin@auburn.edu (X. Qin), weisong@wayne.edu (W. Shi).

https://doi.org/10.1016/j.future.2018.08.014
0167-739X/

The high response latency in transactional storage system caused by write performance cliff is a seriously fault [12]. The buffer management strategy inside SSDs effectively reduces the impact of GC on user I/Os [13].

In buffer management strategies, a great deal of work has been studied. LRU(Least Recently Used) [14] is a popular buffer management strategy constructs a linked list ordering by a page's least recent access time. The page at the end of the list is replaced when the buffer is full. LRU cannot accurately evict the most inactive data while an active data may be replaced out of the buffer. Invalid data pages writing into flash memory can trigger GC. CFLRU(Clean-First LRU) [15] takes into account the asymmetric characteristics in NAND Flash for read and write costs. It divides the linked list into the working region for the nearest access data and the clean-first region for data pages to be deleted. CFLRU improves the accuracy of selecting inactive data in buffer replacement. When there are no clean pages in the clean-first region, CFLRU degenerates into LRU replacement. It ignores the access frequency of dirty pages which occupy the limited buffer space. In addition, a buffer management strategy called GCaR [10] handles user I/Os caused by the conflicts between rewrite operations in flash memory and external I/O requests during GC. When the buffer is full, the data writes into a block of NAND Flash memory which is in the process of GC. Then, the priority of data in the buffer rises and the front data of LRU list is written back into the chip without GC in flash memory. This process avoids the conflicts between the copy-rewrite operation in NAND Flash and I/O requests from an application; thereby improving user response time. GCaR repeatedly checks the state of the chip and also has an impact on user response time.

Previous work [16–18] demonstrates that write performance cliff significantly degrades request response time because of the conflicts between the copy-rewrite operation in GC and user write I/Os in workloads. By studying existing buffer management strategies inside SSDs, we find that the following issues must be taken into consideration when designing an effective approach to improving write performance cliff:

▷ How to store update pages in a fixed number of blocks in flash memory to enhance the efficiency of GC and reduce the number of erased blocks. Write performance cliff caused by GC and its impact on the delay of an I/O request can be mitigated;

▷ How to accurately manage the high-activity data pages in the buffer to avoid frequent traffic into between the buffer and NAND Flash memory. Thus, the delay of the user response time is alleviated by reducing the number of hot-data pages flushing back into NAND Flash memory;

▷ How to aware the invalid data pages in flash memory in real time by the buffer management strategy; thereby performing GC in a distributed mode. This process can improve the impact of block erase in the centralized GC mode on write performance cliff and user response time.

Therefore, we propose a buffer management strategy called ***CalmWPC*** to calm down write performance cliff. CalmWPC mainly includes a cluster-based data organization, a historical data access-based prediction algorithm, and a fingerprint database for the semantic interaction between the buffer and flash memory. The key contributions of this manuscript are listed:

▷ **A cluster-based data organization in the buffer.** The data cluster-based organization groups update data pages into a range of data blocks, which increases the number of invalid data pages in data blocks. This aims to improve GC efficiency and enhance write performance.

▷ **A historical access-based prediction model.** A cluster with high activity will stay in the buffer for a long period by this prediction model. A data cluster has the same maximum number of data pages as the corresponding block in flash memory.[1] At an appropriate time, an inactive cluster writes back into NAND Flash memory. The utilization of the buffer increases while the number of rewrites of invalid pages reduces, which improves write performance cliff.

▷ **A semantic interaction-aware fingerprint database.** The fingerprint database including semantic messages bridges the buffer and flash memory. A fingerprint database is composed of many fingerprint sets, and each fingerprint set contains many fingerprint message which relates with data blocks in flash memory It makes a data cluster aware the invalid pages of the corresponding data block in flash memory. This can avoid inactive data clusters to be frequently written into flash memory. When the number of update pages in an inactive cluster reach the threshold, the data cluster flushes into flash memory, and the original data block is added to the GC queue when valid data pages in the original data block are rewritten into a new one. The distributed GC is applied to improve write performance cliff and shorten user response time.

The data cluster is the basic data structure for the prediction algorithm and the fingerprint database. The update pages concentrate on a fixed range of addresses in flash memory, and many invalid data pages exist in the same block, which reduces valid pages rewrite operations and enhances GC efficiency. The probability of adversary interference between the user write I/O and GC-induced I/O traffic decreases during GC. By means of the prediction algorithm, a data cluster can obtain its activity value which can classify the data cluster into the high- or low-activity data cluster sets. The prediction algorithm reduces the number of the high-activity data cluster which will be written back into flash memory. The fingerprint database for write/update operations has semantic messages that bridge the buffer and flash memory. The fingerprint database detects whether a user I/O is an update operation or not in the buffer. CalmWPC real-time compares the number of invalid pages in a data block with the threshold. When the number of update data pages reaches the threshold, the cluster flushes into its corresponding data block in NAND Flash memory. These valid data pages in the data block are rewritten into a new one, and then, the original data block is added to the GC queue. When the chip inside an SSD is idle, the original data block performs GC to improve the utilization of the user space in NAND Flash memory. GC executes in a distributed mode rather than the centralized one to reduce write performance cliff and avoid the high user response time.

This paper is organized as the following. Section 2 mainly introduces the background. Section 3 provides the theoretical model of CalmWPC, and the algorithm in detail. Section 4 describes the effect of the system experiment. Section 5 demonstrates the related work. We summarize our work in this manuscript in Section 6.

2. Background

With the rapid development of big-data applications [19], the scale of storage and computing for these applications faces great challenges [20]. Currently, HDDs as the main storage devices have lower performance compared with CPU computing power. Due to the inherent characteristics, the HDD-based storage performance becomes the bottleneck for the whole system [21] under random-write-intensive workloads. Therefore, NAND Flash-based solid state disks (SSDs) are popularized to enhance the storage performance. Compared with the HDDs, SSDs have the advantage of high performance and low power consumption [3–5,22].

[1] Note that the data cluster is in the buffer and the data block exists in flash memory.