Don't Let RAID Raid the Lifetime of Your SSD Array

Sangwhan Moon
Texas A&M University

A. L. Narasimha Reddy
Texas A&M University

Abstract

Parity protection at system level is typically employed to compose reliable storage systems. However, careful consideration is required when SSD based systems employ parity protection. First, additional writes are required for parity updates. Second, parity consumes space on the device, which results in write amplification from less efficient garbage collection at higher space utilization.

This paper analyzes the effectiveness of SSD based RAID and discusses the potential benefits and drawbacks in terms of reliability. A Markov model is presented to estimate the lifetime of SSD based RAID systems in different environments. In a single array, our preliminary results show that parity protection provides benefit only with considerably low space utilizations and low data access rates. However, in a large system, RAID improves data lifetime even when we take write amplification into account.

1 Introduction

Solid-state drives (SSDs)[1] are attractive as a storage component due to their high performance and low power consumption. Advanced integration techniques such as multi-level cell (MLC) have considerably dropped cost-per-bit of SSDs such that wide deployment of SSDs is feasible. While their deployment is steadily increasing, their write endurance still remains as one of the main concerns.

Many protection schemes have been proposed to improve the reliability of SSDs. For example, error correcting codes (ECC), log-like writing of flash translation layer (FTL), garbage collection and wear leveling improve the reliability of SSD at the device level. Composing an array of SSDs and employing system level parity protection is one of the popular protection schemes at the system level. In this paper, we study striping (RAID0), mirroring (RAID1), and RAID5.

RAID5 has improved the lifetime of HDD based storage systems for decades. However, careful decisions should be made with SSDs when the system level parity protection is employed. First, SSDs have limited write endurance. Parity protection results in redundant writes whenever a write is issued to the device array. Unlike HDDs, redundant writes for parity update can severely degrade the lifetime of SSDs. Parity data consumes device capacity and increases the space utilization. While it has not been a serious problem in HDDs, increased space utilization leads to less efficient garbage collection which in turn increases the write workload.

[1] We consider MLC SSDs in this paper, unless otherwise mentioned.

Many studies have investigated SSD based RAID systems. The notable study [1] points out the pitfalls of SSD based RAID5 in terms of performance. They discuss the behavior of random writes and parity updates, and conclude striping provides much higher throughput than RAID5. We consider the impact of write workload on reliability. Previous studies [2, 3] have considered different architectures to reduce the parity update performance penalty. We focus on the problem of random and small writes resulting in frequent parity updates. Recent study [4] shows that randomness of workload is increasing with the advent of big data analytics and virtualization.

This paper focuses its attention on the reliability of an array of MLC SSDs. In this paper, we explore the relationship between parity protection and the lifetime of an SSD array. The paper makes the following significant contributions:

- We analyze the lifetime of SSD taking benefits and drawbacks of parity protection into account.
- The results from our analytical model show that RAID5 is less reliable than striping with a small number of devices because of write amplification.

Sec. 2 provides background of system level parity protection and write amplification of SSDs. Sec. 3 explores SSD based RAID. Sec. 4 builds a reliability model of SSD based RAID. Sec. 5 shows evaluation of our statistical model. Sec. 6 concludes this paper.

2 Backgroud

We categorize protection schemes for SSDs into two levels: device level protection and system level protection. Device level protection includes ECC, wear leveling, and garbage collection. System level protection includes RAID5 and mirroring. In this paper, we will mostly focus on system level protection.

2.1 System Level Protection

In many cases, device level protections are not enough to protect data. For example, when the number of bit errors exceeds the number of correctable bit errors using ECC, data in a page may be lost without additional protection mechanisms. A device can fail due to other reasons such as the failure of device attachment hardware. In this paper, we call the former as a page error and the latter as a device failure. In order to protect device failures, system level parity protection is employed.

RAID5 is popular as it spreads workload well across all the devices in the device array with relatively small space overhead for parity protection. The group of data blocks and the corresponding parity block is called a

page group. RAID5 is resilient to one device failure or one page error in a page group.

Mirroring is another popular technique to provide data protection at the system level. Two or more copies of the data are stored such that a device level failure does not lead to data loss unless the original and all the replicas are corrupted before the recovery from a failure is completed. When the original data is updated, the replicas have to be updated as well. Read operations can be issued to either the original or the replicas at the system level. When a device is corrupted, the paired devices are used to recover the failed device.

2.2 Write Amplification

Protection schemes for SSD often require additional writes and those writes in turn reduce the reliability of SSDs. Since higher write amplification can reduce the lifetime of SSD severely, protection schemes should be configured carefully to maximize the lifetime improvement while minimizing write amplification. Write amplification severely degrades reliability, since the reliability is highly dependent on the number of writes done at the SSDs. Main sources of the write amplification are discussed in this section.

Recovery process In most of the recovery processes, at least one write is required to write a corrected page. ECC can correct a number of errors in a page simultaneously with one write. For instance, fixing a bit error in a page takes one redundant write, while fixing ten bit errors in a page also needs one additional write.

Our previous work [5] suggested threshold based ECC (TECC) to reduce the write amplification from frequent recovery by leaving bit errors in a page until it accumulates a certain number of bit errors. TECC can drastically reduce the write amplification from ECC. It is noted that ECC based write amplification is a function of read workload unlike other sources of write amplification.

Garbage collection NAND flash memory is typically written in a unit of a page, 4KB, and erased in a unit of a block, e.g., 512KB. It does not support in-place update and accumulates writes in a log-like manner. In such a log structured system, internal fragmentation and the garbage collection process to tidy the fragmented data are inevitable. The garbage collection process moves valid pages from one place to another place and this results in increasing the number of writes issued to the device. Write amplification due to garbage collection is strongly dependent on the space utilization of the SSD. When the SSD is nearly full, garbage collection initiates quicker and results in being less efficient since a larger fraction of the blocks are still live. Recent study [6] reveals that the efficiency of garbage collection is also dependent on the hotness of data. Since hot data tends to be more frequently invalidated, garbage collection is more efficient when hot workload concentrates on a small portion of data.

3 SSD based RAID

Our analysis is based on an architecture where a RAID controller operates on top of a number of SSDs. As a result, the set of pages within a page group have a constant logical address (before FTL translation) within the device. As the pages are written, the actual physical address of pages within the device change because of FTL translation. However, this does not impact the membership of the page group, based on the logical addresses. When the number of page errors in a page group exceeds the number of correctable page errors by RAID, the page group fails and the storage system loses data.

In RAID5, for a small write, the data block and the parity block need to be updated, potentially resulting in a write amplification factor of 2. However, when a large write that spans a number of devices is issued, the parity block can be updated once for updating N-1 data blocks with the help of a parity cache [2], where N is the number of devices in the device array, resulting in a write amplification factor of N/(N-1). Depending on the workload mixture of small and large writes, the write amplification will be somewhere in between. In a mirroring system, the write amplification is 2 regardless of the size of the write request.

Parity blocks increase space utilization. Suppose that 120GB of data is stored in four 80GB SSDs, RAID5 stores 30GB of data and 10GB of parity in each device while striping stores only 30GB data per device. The increased amount of data results in less efficient garbage collection and more write amplification, which decreases the lifetime of SSDs.

4 Lifetime Model

Our lifetime model is based on a Markov model of [5] which analyzes the relationship of single SSD's reliability and device level protection. A number of symbols used in our model are shown in Table 1.

A number of studies [7, 8] have investigated the MLC bit error behavior of flash memory. There are different sources of bit error of flash memory: read disturb, data retention failure, and write error. These studies model the bit error rate as an exponential function of the number of program/erase cycles (P/E cycles) the cell has gone through. We employ the data from the measurement study of [8] to model the change of bit error rate. Table 2 shows parameters of the employed model.

Uncorrectable Page Error Rate We employ a series of Markov models in Fig. 1 to build a model of reliability of a page. We can estimate the uncorrectable page error rate at each slot using a steady state analysis of the Markov chain. From the analysis, the probability of reaching the

Symbols	Description
S	the number of bits in a page
K	the number of correctable bits by ECC per page
x	P/E cycles
$\lambda(x)$	raw bit error rate
i	(state) the number of bit errors in a page
$g(i,x)$	garbage collection access rate
μ	page access rate
μ_r	page read rate
F_E	(state) page error: ECC cannot recover the page
v_i	(state) a page is corrupted
d_i	(state) a device fails
N	the number of devices
u_s	space utilization of SSD
T	the number of pages in an SSD
$\nu(x)$	uncorrectable page error rate
d	device failure rate
ξ	page group recovery rate
η	device recovery rate
F_R	(state) data loss: RAID cannot recover the system
$\sigma(x)$	data loss rate at system level
$p(x)$	data loss probability induced by $\sigma(x)$
t_w	time to write whole system once
P_i	the prob. of staying at state i in a Markov model
w	the number of writes issued per second.

Table 1: List of symbols used in statistical models

Error Type	A	B
Read disturb (per read)	3.11e-7	2.17e-4
Data retention (per month)	3.30e-6	1.83e-4
Write error (per write)	1.09e-7	3.01e-4

Table 2: Raw bit error rate $\lambda(x) = Ae^{Bx}$

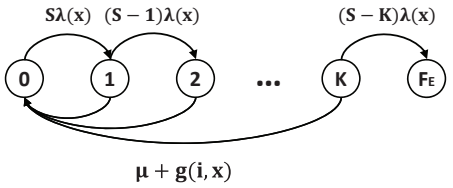

Figure 1: A Markov model of page error rate

state F_E can be obtained. Detailed presentation is omitted due to lack of space.

Mean Time to Data Loss We employ the mean time to data loss (MTTDL), which is one of the popular metrics in reliability analysis. The MTTDL is defined as the expected time to encounter the first data loss: $\text{MTTDL}=\left(\sum_{j=1}^{\infty}\left(jp(j)\prod_{i=1}^{j-1}(1-p(i))\right)\right)\cdot t_w$. The data loss rate $\sigma(x) = u_s \cdot T \cdot v(x)$ for single SSD.

The reliability of SSD is highly dependent on the number of writes it has serviced. We employ a detailed Markov model in Fig. 2 considering replaced devices after device failure recovery. The replaced devices have experienced relatively lower number of writes.

Two sources of failure are considered in this model. Bit errors accumulate and result in a page error. The entire device failure can lead to the failure of all the pages in the device. The data loss rate $\sigma(x)$ is the rate reaching state F_R. For N device striping, $\sigma(x) = Nu_sT \cdot v(x)$.

Under an assumption that each device pair is independent in a mirroring system, we can build a Markov model for the device pair, and then extend it to cover the entire system. The pair fails with failure rate $\rho(x)$, and the mirroring system fails at the rate of $\sigma(x) = N/2 \cdot \rho(x)$ where N is the number of devices in the mirroring system.

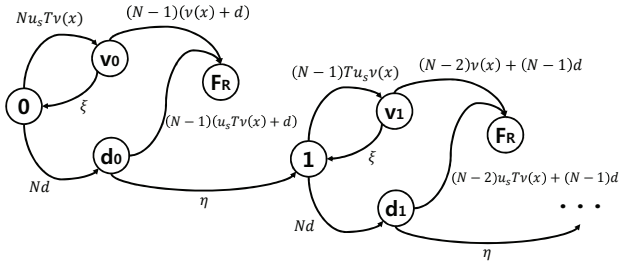

Figure 2: A Markov model of RAID5 system

4.1 Write Amplification

Write amplifications are caused by garbage collection, ECC recovery, and parity update which are denoted as $\alpha_{gc}, \alpha_{rcv}$, and α_{parity}. We exploited the result of [6] to estimate garbage collection rate $g(i,x)$ which is dependent on write workload, and $\alpha_{gc} = \sum_{i=0}^{K}(g(i,x)\cdot P_i)/w$. And $\alpha_{rcv} = \left(\mu_r \sum_{i=1}^{K} P_i\right)/w$. α_{parity} depends on workload characteristics and is in the middle of N/(N-1) and 2.

Device level ECC recovery and system level parity update are independent. Block level recovery does not require parity update and vice versa. But garbage collection depends on the amplified writes from parity updates and ECC recovery: $\alpha = (\alpha_{rcv} + \alpha_{parity} - 1)\cdot\alpha_{gc}$. This write amplification is emulated by changing sampling of data loss rate $\sigma(x)$.

5 Evaluation

The lifetime of an SSD array is evaluated in different environments. Various aspects of SSD array are explored.

5.1 Simulation Environment

We exploited bit error rate mainly from [8], specifically 3x nm memory. We assume that 61-bit correctable ECC for a 4KB page is employed. We assume a constant device failure rate of 3% [9] over its lifetime. Each SSD has a capacity of 80GB.

30GB of data is kept under 125MB/s of workload, with 3:1 read:write ratio, per device as default. The actual space utilization can be more than 30GB due to parity or replication.

TRIM command issued by file system indicates to SSD which data is invalidated. We assume that the TRIM command is exploited with zero overhead.

The lifetime varies by vendor and model of SSD. We normalize the lifetime results to the lifetime of a single SSD in a default environment. We call the normalized lifetime as relative MTTDL.

5.2 Review of Analysis of Single SSD

Space utilization and the write amplification from garbage collection are strongly related to the lifetime.

Space util.	0.1	0.3	0.5	0.7	0.9
R. MTTDL	1.0993	1.0620	0.8966	0.6243	0.26056

Table 3: Lifetime of an SSD v.s. space utilizations.

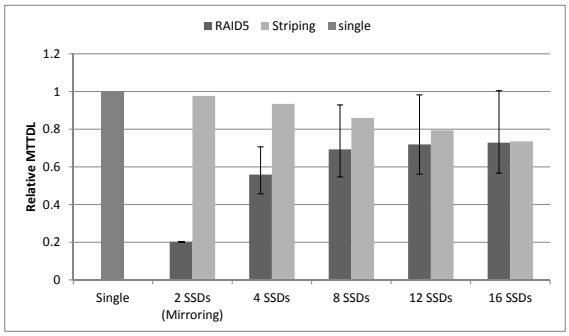

Figure 3: Lifetime of different number of SSDs

Table 3 shows the relationship between space utilization and the lifetime of a single SSD. It shows an interesting point: the expected lifetime does not decrease linearly as space utilization increases. When we increase space utilization from 0.1 to 0.5, the amount of data is five times the original data, with only 18% loss of lifetime. If we increase space utilization from 0.5 to 0.9, in contrast, the lifetime decreases by about 70%.

5.3 Is RAID5 Superior to Striping?

We compare the lifetime of RAID5 to the lifetime of striping in this section. Since the write amplification from parity update varies depending on small vs. large writes, we describe the lifetime of RAID5 as a range from its minimum to maximum achievable.

We first analyze the impact of write amplification to the lifetime of SSD based arrays. We vary parameters to find when RAID5 is superior to striping and when RAID5 may not be so beneficial in this section.

The number of devices Fig. 3 compares the lifetime of different systems with varying number of SSDs. It brings out many interesting implications. First, mirroring, at 2 SSDs suffers extremely from higher write amplification. Its write amplification from parity update (or replica copy) is always 2. Its space utilization is twice as that of striping on 2 SSDs. The lifetime of RAID5 systems grows with the number of devices. The results shows that with less than 8 SSDs the overhead from additional writes from parity overwhelm RAID5's reliability benefits.

The amount of data Since the space utilization of RAID5 is higher than striping due to parity, it is more sensitive to space utilization than striping. This tendency is shown in Fig. 4. As we increase the amount of data, the lifetime of RAID5 drastically decreases and eventually its maximal lifetime is less than striping. This implies that the total amount of data should be less than a certain amount otherwise RAID5 will not be beneficial in terms of reliability. RAID5 is shown to be competitive when the amount of data is less than 240GB or

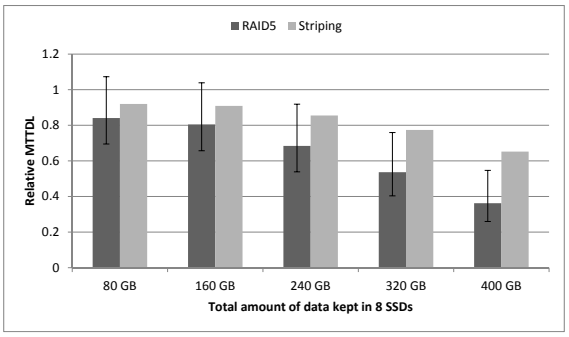

Figure 4: Lifetime of 8 SSDs of different amount of data

space utilization is less than about 40%. The 400GB of data is equivalent to 71% space utilization or 40% over-provisioning. Considering that recent SSDs typically provide over-provisioning from 7% to 20% (93% to 83% of space utilization), additional measure should be taken to 1) increase over-provisioning considerably and/or 2) reduce write amplification further.

Advanced techniques In Fig. 3 and Fig. 4, it is notable that the lifetime gain is not linearly increasing as write amplification decreases. This is because write amplification from various sources are not independent. When the number of devices increases, for example, write amplifications from both parity update and garbage collection reduce at the same time. Moreover, when parity update overhead decreases, garbage collection rate is also decreased and it results in less write amplification from garbage collection.

Many techniques have been recently proposed to reduce the write amplification. Among them, we evaluated in Fig. 5, threshold based ECC (TECC) which reduces the write amplification from ECC recovery. As we expected, lifetime increases considerably. With TECC, write amplification from parity update has more impact on the lifetime. As explained in Sec. 4.1, reducing write amplification from ECC recovery makes the influence of write amplification from parity update relatively larger.

Hotness of blocks reduces write amplification from garbage collection. As, seen in Fig. 5, hotness of data (when 80% of data is concentrated on 20% of space) results in slightly higher lifetime.

Workload characteristics We consider other parameters such as read:write ratios and data access rates in Fig. 5. The workloads that are less write intensive result in SSDs that wear out at a slower rate while constant device failure rates contribute more to data loss. Since RAID5 is robust to device level failure, its lifetime grows more than expected. The 62.5MB/s workload has more writes than R:W=9:1 workload but it has lower read bandwidth which in part converts to redundant writes from frequent ECC recovery.

Device failure rate We evaluated the lifetime of RAID5 with higher failure rate (annually 5%) in Fig. 6. In

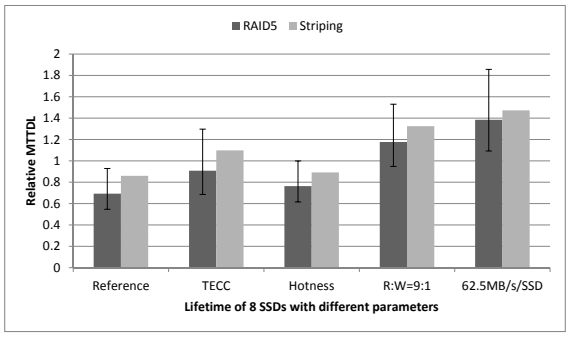

Figure 5: Lifetime of 8 SSDs with different parameters

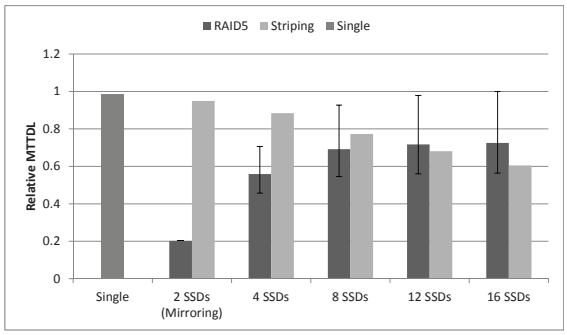

Figure 6: At higher annual device failure rate (5%)

the figure, when the number of devices is more than 8, RAID5 is better in lifetime on an average than striping. This result shows that RAID5 is useful with larger device failure rate and larger number of devices in the array. The larger number of devices contributes to larger failure rate at the system level as well as to smaller overhead of parity in write workload and capacity.

Scalability Our evaluation results so far have shown that RAID5 may not always be beneficial in improving the lifetime compared to striping when we consider a single array of 4-16 SSDs. However, when we scale out those results to many devices, we see a totally different story. We compare the lifetime of a system consisting of many 4-SSD arrays to the lifetime of striping in Fig. 7. The results show that striping is very poor at scaling out since only one failure in tens of devices results in data loss. We cannot argue to compose large-scale storage systems without parity protection at the system level, but our results show that RAID5 is not universally beneficial, especially with small number of devices.

Most of our evaluation results imply that write amplification is key to understanding the reliability of SSD arrays:

(1) Write amplification can be rather harmful to the reliability of SSD arrays. They should be carefully examined before employment.

(2) Many factors can change write amplification like space utilization, hotness and randomness of workload. More efficient system level protection techniques need to be designed taking write amplification into account.

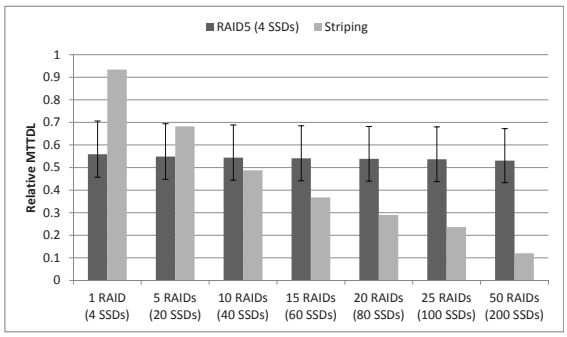

Figure 7: A series of 4-SSD RAIDs v.s. striping

More research is needed. Other evaluations such as MTTDL with different write sizes, MTTDL with other storage systems (e.g. RAID6), and the cost-of-ownership of different storage systems will be studied in the future. In addition, we plan to validate our analytic model with simulations in the future.

6 Conclusion

This paper analyzed the relation between parity protection and the lifetime of SSD arrays. Parity update increases write workload and space utilization which can severely degrade the reliability of SSD arrays.

According to our analytical model and evaluation, we show that RAID5 is conditionally better in lifetime than striping due to the overhead of parity update. Different factors such as the number of devices and the amount of data are explored, and the results imply that RAID5 is not universally beneficial in improving the reliability of SSD based systems.

Acknowledgments

We thank our shepherd Hakim Weatherspoon for his help in improving the presentation of the paper.

References

[1] N. Jeremic, G. Muhl, A. Busse, and J. Richling, "The pitfalls of deploying solid-state drive RAIDs," in SYSTOR'11.

[2] K. Greenan, D. Long, E. Miller, T. Schwarz, and A. Wildani, "Building Flexible, Fault-Tolerant Flash-based Storage Systems," in HotDep'09.

[3] G. Soundararajan, V. Prabhakaran, M. Balakrishnan, and T. Wobber, "Extending SSD lifetimes with disk-based write caches," in FAST'10.

[4] V. Tarasov, D. Hildebrand, G. Kuenning, and E. Zadok, "Virtual Machine Workloads: The Case for New NAS Benchmarks," in FAST'13.

[5] S. Moon and A.L.N. Reddy, "Write amplification due to ECC on flash memory or leave those bit errors alone," in MSST'12.

[6] P. Desnoyers, "Analytic modeling of SSD write performance," in SYSTOR'12.

[7] L. Grupp, A. Caulfield, J. Coburn, S. Swanson, E. Yaakobi, P. Siegel, and J. Wolf, "Characterizing flash memory: Anomalies, observations, and applications," in MICRO'09.

[8] H. Sun, P. Grayson, and B. Wood, "Quantifying Reliability of Solid-State Storage from Multiple Aspects," in SNAPI'11.

[9] K. Thomas, "Solid State Drives No Better Than Others, Survey Says," [Avaliable] www.pcworld.com/article/213442