

RAID-6Plus: A Fast and Reliable Coding Scheme Aided by Multi-failure Degradation

Ming-Zhu Deng^(✉), Yang Ou, Nong Xiao, Song-Ping Yu, Wei Chen, Zhi-Guang Chen, and Fang Liu

State Key Laboratory of High Performance Computing, College of Computer,
National University of Defense Technology, Changsha 410073, China
dk_nudt@126.com

Abstract. Existing triple-failure-tolerant codes assume that failures are independent and instantaneous. Such assumptions overlook the underlying mechanism of multi-failure occurrences and ignored the effect of reconstruction window. These codes are not adapted to the occurrence pattern of failure in real-world applications. As a result, the third parity drive is almost idle as it set to handle the triple-failure scenario only with lower-level failure situations unattended. Furthermore, the problem of single failure rebuild deteriorates with the increasing disk capacity, and the system's reliability will decrease with user experience impaired. Aiming at these problems, a fast reconstructable coding scheme extended from RAID-6 has been developed in this study. RAID-6Plus maintains a smaller reconstruction window by recoding the third parity drive. Existing codes provide absolute reliability for triple failures via full combinations. As a contrast, RAID-6Plus employs short combinations which are able to greatly reuse overlapped elements during reconstruction to remake the third parity drive. The short combinations shorten the reconstruction window of single failure, which avoids multi-failure overlapping in the reconstruction window. The capability of multi-failure degradation provides RAID-6Plus with (1) a better system performance comparing to RTP and STAR and (2) an enhanced reliability comparing to RAID-6.

Keywords: Reconstruction window · Failure mode · Multi-failure degradation · Flexible reliability

1 Introduction

RAID-6 is conventionally the de facto default configuration of storage systems for its reliability. However, modern storage systems are more and more exposed to higher level of vulnerability in the big data era. Data reliability has unprecedentedly become a critical issue to be addressed. In terms of device level failures, Huang and Xu proposed STAR code [1] to protect a storage array from triple failures. Goel and Corbett extended RDP to RTP [2] with a faster coding algorithm for handling triple failures. Similar codes include RS codes and generalized EVENODD [3]. Jain et al. introduced redundancy between RAID groups by mirroring or replication at the

expense of a higher overhead [4]. Codes along this direction can generally provide satisfactory solutions to triple failures, but the following problems are still pending especially the lack of flexibility for enabling reliability:

- Existing codes assume that failures are independent and instantaneous and occurrences of failures conform to the exponential distribution [5, 6]. This ideal assumption does not apply in the fault pattern of modern storage systems [7]. Furthermore, those codes are not designed to support multi-failure degradation, which aims to convert a higher-level multi-failure into separate low-level multi-failures or single failures with a shorter reconstruction window.
- Existing codes largely ignore the pattern of failure occurrences in practice. For example, 99.75 % of recoveries are due to single disk failures [16], while triple whole-disk failures are rare. However, the third parity drive in RTP is set to handle the triple-failure scenario only with single failure rebuild unattended. The third parity drive is then almost wasted.
- Existing codes focus on whole-device level failures while mixed-fault modes are more common in practice [8]. For example, when a fault consisting of two erasures and a sector error occurs, all of the three parity drives must be used. This directly overkills the effects of the third parity drive.
- As throughput is dwarfed by capacity [9], the reconstruction windows have grown exponentially from minutes to hours or even days in practice. This leads to a severe decrease of system performance and poor user experiences.

There exists a pressing need for a coding design to support multi-failure degradation and a smaller reconstruction window to deliver data reliability in a more flexible manner. In this study, we developed a fast reconstructable coding scheme with shorter reconstruction window, namely RAID-6Plus. The proposed coding scheme is an extension of RAID-6 [10] at the expense of three parity drives. RAID-6Plus employs short combinations which are able to effectively reuse overlapped elements during reconstruction to remake the third parity drive. This design shortens the reconstruction window of single failures by minimizing the total number of data reads. The possibility of multi-failure overlapping in the reconstruction window is therefore significantly diminished. The capability of multi-failure degradation provides RAID-6Plus with (1) a better system performance comparing to RTP and STAR and (2) an enhanced reliability comparing to RAID-6. RAID-6Plus balances reliability, performance and update penalty, which aims at practical uses in modern disk systems, flash-based systems, and even hybrid storage system on any array codes.

RAID-6Plus was then evaluated against the RTP and STAR codes. A mathematic analysis indicated that RAID-6Plus could achieve speedups of 33.4 %, 11.9 %, 47.7 % and 26.2 % comparing to RTP with conventional rebuild, RTP with optimal rebuild, STAR with conventional rebuild and STAR with optimal rebuild respectively. Furthermore, RAID-6Plus is orthogonal with some previous work on reconstruction speedup [10–13] and can be integrated together to further shorten reconstruction window.

The main contributions of this study are as follows:

- We developed a new XOR-based coding scheme with shorter reconstruction window and lower risk of multi-failure with no additional cost incurred comparing to RTP

and STAR, which enables a balanced solution with flexible reliability and better system performance.

- We design a novel redundancy scheme to minimize data reads in reconstruction scenarios. The redundancy scheme is generic and could be applied to other codes.
- To the best of our knowledge, RAID-6Plus is the first to extend given codes from the perspective of accelerating single failures instead of providing reliability boundary. The design is better adapted to the occurrence pattern of failures in practice.

The remainder of the paper is structured as follows. Section 2 motivates RAID-6Plus. Section 3 presents the design of RAID-6Plus. Finally, we evaluate the performance of RAID-6Plus in Sect. 4 with conclusions in Sect. 5.

2 Backgrounds and Motivation

In this section, we describe the different failure modes and recovery methods in RAID systems and introduce the concept of multi-failure degradation in reconstruction window. This motivates the need to design RAID-6Plus, which exploits multi-failure degradation mechanism.

2.1 Failure Modes in Disk and SSD

Multiple researches on the massive disks have all shown that mainstream disk drives have device failures or whole-disk failure [14] and sector failures. Device failure refers to the loss of the whole disk, which is mainly caused by hardware problems or manual misconducts. Sector failures could be caused by diverse reasons such as software glitches and so on [15]. All this failures directly cause data unavailability.

More specifically, newest findings by Ao ma reveal that despite infant mortality, multi disks tend to fail at similar age. That means that not only single failure happens at the device level, but also multiple devices may fail almost simultaneously, calling for higher reliability care than RAID-6. Other statistics show among all device failures, single failure accounts for the majority (99.75 %), and double failures is not eligible (roughly 8 %) and should be taken care of [16]. Meanwhile, often the cases are single failure coupled with other sector failures rather than multi whole-disk failures [8]. And sector errors are not rare and increase with time [14].

In terms of correlation between whole-disk failure and sector errors, reference [14] asserts whole-disk failure can be viewed as the consequence of accumulated sector error and uses the number of reallocated sectors to characterize probability of whole-disk failure. Further, the longer a functioning device undergoes, the higher probability of device failure could be. In other words, other failures could happen in ongoing process of failure recovery, aggravating system reliability and making it much more vulnerable [17]. In short, single failure is of vital importance to reduce window of system vulnerability.

Though there have not been any investigation published on massive SSD, SSD's failure modes must be similar while differ in some of its own vulnerability features, like its inborn limited endurance issues and wearing over time [18].

2.2 Growing Reconstruction Window for Single Failure

Basically, given the same rotational speed, reconstruction window for a single disk varies according to different disk capacity [19]. Therefore, through simple calculation, reconstruction windows has also grown hugely along with device capacity, leading to easier opportunity for potential data loss [17].

On the contrary, the shorter reconstruction window is, more failures with higher level can be avoided by degraded into low-level failures. Current reconstruction window is unbearably large. For example, it would normally take more than two hours to reconstruct the data of a 500 GB disk. This not only aggravates user experience for longer wait, but also puts the whole system into higher risks of data loss. Therefore, the urgent need to shorten reconstruction window is practical and would be welcome.

2.3 Motivation

Traditional codes with higher fault-tolerance, like RTP and STAR codes are originally designed for triple whole-device failures. Unfortunately, statistics have shown probability of single failure overwhelmingly accounts for most while triple whole-device failures is relatively rare, thus making current RAID-7 system with triple parity drives wasteful. Additionally, mixed-fault modes exhibited in modern storage systems, overkills the solution of those codes with higher reliability boundaries. In short, current RAID-7 system with triple parity coding is unpractical and needs to deliver more flexible reliability.

Further, reconstruction window is exponentially increasing with device capacity, thus worsening user experience and leaving larger space of system vulnerability and data loss. More notably, current coding schemes for triple-failure are unable to shorten reconstruction window.

Therefore, all these factors above motivate me to extend RAID-6 codes another way to shorten reconstruction window to provide flexible reliability and make it more practical.

3 System Design

In this section, we base on RDP code to illustrate our RAID-6Plus. First of all, we introduce the optimal way of RDP for single failure rebuild. Then, in RAID-6Plus, we keep the P disk and Q disk as it is, and present our new coding algorithm for the third parity drive X to reduce reconstruction window.

3.1 Optimal Reconstruction for Single Failure in RDP

In RDP, we have two kinds of parity drives, where P drive means slope 0 and Q for slope 1. whenever any single data disk fails, the conventional way to reconstruct a failed disk is merely using P drive while the optimal way is using equal number of parities from P and Q drives, maximizing the overlapping data, as shown in (Fig. 1) [10].

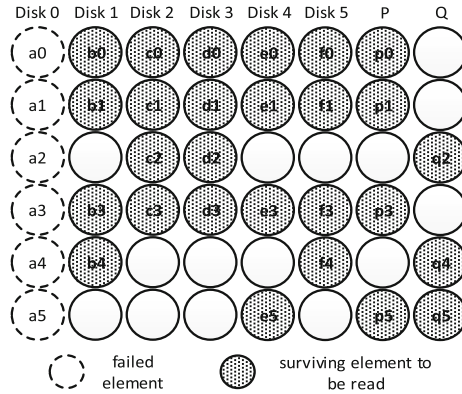


Fig. 1. The optimal reconstruction sequence of single failure in RDP. The hybrid use of equal numbers of P parities and Q parities maximizes the overlapping data for single failure rebuild.

Table 1. Feature comparison of three coding ways for X drive

Coding methods	Element example	Involved element	Merit	Shortcoming
Mirroring of single disk	a1	1	simple and can maximize reduction of reconstruction window for specific drive	Only cover replicated drive, not able to cover other drives, causing imbalance and fluctuation
Short combination	a1 + b1	2	In between	
Full combination	a1 + b1 + c1 + d1 + e1	5	Maximize fault-tolerance for the whole system	Unable to reduce reconstruction window

3.2 Redundancy Coding in RAID-6Plus

In order to provide solid and flexible reliability against multi-failure, RAID-6Plus uses three redundancy drives as other codes for triple-failure do, but in a new and more practical way. Actually, there is a reasonable compromise between reliability level and system performance in RAID-6Plus, where on the one hand, we maintain

a basic reliability guarantee to accommodate non-negligible double-failure, while on the other hand, reconstruction window is shortened to degrade failure and reduce user wait.

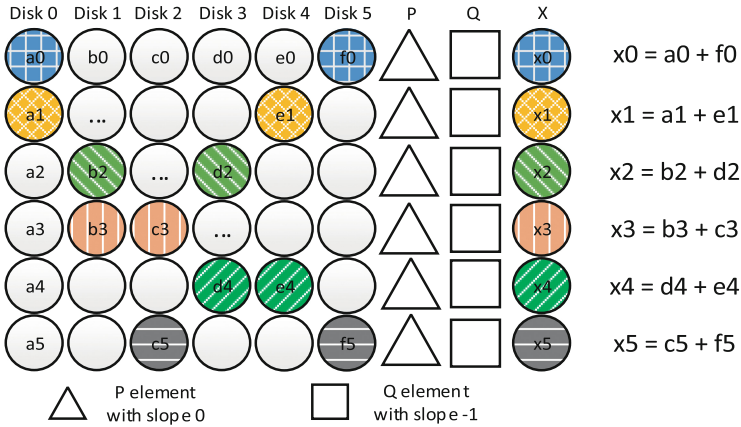


Fig. 2. Encoding for X drive in RAID-6Plus. Any new parity in the third parity drive X is the XOR of some two data elements. There are two data elements of each data drive to be covered in X drive. And those data pairs or tuples to construct X parity is specially chosen to satisfy fast reconstruction.

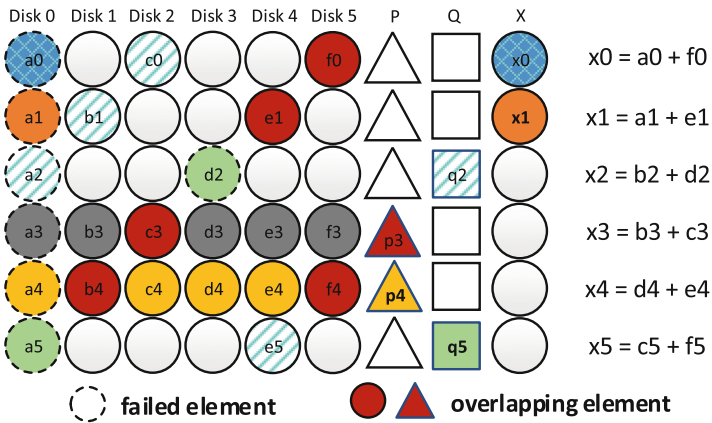


Fig. 3. Rebuilding single failure with X elements. a0 and a1 are recovered by x0 and x1, a2 and a5 are rebuilt with the help of q2 and q5 while a3 and a4 are fixed with p3 and p4.

Of all the three redundancy drives, the first two are devoted to maintaining a lower bound of reliability, which we denote as “base reliability”, given RAID-6 is widely deployed in diverse storage system for double-failure concern. Therefore, we have the exact P drive and Q drive in RAID-6Plus from RDP to keep a base reliability guarantee for double-failure.

The remaining redundant X drive is dedicated to reducing reconstruction window for single failure of any data drive, which is in charge of user access. Therefore, the key lies in the redundancy coding for X drive. Conventionally, there are three ways for X drive coding: (1) Mirroring of a single data disk; (2) Short combination; and (3) Full combination. The features of the three coding ways are so obvious that we summarize them in Table 1.

In essence, the three ways of coding for X parity elements vary in number of element involved in the encoding process. Mirroring has a length of only one, suggesting any X element is a replica of some element in other drives. Full combination has the same length of any element in P or Q drive, implying any element in X is the combination of many elements in data drives while Short combination is in between.

In order to reduce data reads as much as we can, we employ short combination with some 2 elements involved to encode for X parity. The thought behind is to find overlapping elements as many as possible on the basis of optimal reconstruction for single data disk in RDP and combine any two of them for the concern of data disk coverage, as shown in Fig. 2.

We use the example in Fig. 2 to illustrate. In our coding for X drive, nearly all the data drives are covered. For example, we have $a0$ and $x1$ of disk #1 in the short combination, which will be used in the reconstruction of disk #1. In other words, disk coverage in some way guarantee the balance of effect distribution on multi disks.

We use the example in Fig. 3 to illustrate. In our coding for X parities, nearly all the data drives are covered. For example, we have $a0$ and $a1$ of disk #0 respectively in $x0$ and $x1$, which will be used in the reconstruction of disk #0. In other words, disk coverage in some way guarantees the even and balanced distribution of speedup effect on multi disks.

And those short combinations in X drive constructed by data pairs are specially chosen to satisfy fast reconstruction.

3.3 Reconstruction in RAID-6Plus

In terms of single erasure reconstruction, for example, if $disk\#0$ fails, reconstruction with the participation of related short combinations in X will occur.

As shown in Fig. 3, we use $x0$ and $f0$ to recover $a0$, $x1$ and $e1$ for $a1$. And respectively with the help of $p3$ and $p4$, $a3$ and $a4$ are reconstructed in the direction of slope 0 while $a2$ and $a5$ are respectively recovered from slope -1 with $q2$ and $q5$. In total, there are 22 elements needed to be read, comparing with 27 element reads of RDP optimal recovery.

In the views of double failures, RAID-6Plus maintains system reliability in two ways. First and foremost, RDP is actually included in RAID-6Plus, therefore there will be no data loss whenever any double failures happen.

Additionally, with shorter reconstruction window, some double-failure situations previous in traditional RAID-6 system could be converted to independent single failures, thus eliminating vulnerability undergone by the system. Also, the same way of using short combinations in X can be applied to save data reads for double-failure scenarios.

In fact, triple failures happen at an extremely tiny probability, therefore the third parity drive in traditional codes only exist for extreme cases. And according to Reliability Equation in [20], RAID-6Plus could convert a portion of triple-failure situations

in traditional coding schemes to lower possibility, leaving unconvertible triple failures at a negligible level.

4 Performance Analysis

In this section, we analyze the performance of RAID-6Plus, with RTP and STAR codes. The analysis comes in parity computation complexity, update penalty, reconstruction cost, and storage ratio. Further, we propose Q metric to denote the induced performance improve per cost to validate our RAID-6Plus. The basic data matrix for STAR is $(p - 1)p$ and $(p - 1)(p - 1)$ for RAID-6Plus and RTP.

4.1 Parity Computation Complexity

We compare parity computation complexity of different codes by XORs per element. The total XORs can be computed as the sum of XORs for different parity types. For example, the number of XORs for writing a stripe in RAID-6Plus is consisted of XORs for row parity, diagonal parity and X parity, making

$$\begin{aligned}
 A_{RAID-P} &= \frac{XOR_{row} + XOR_{dia} + XOR_x}{Blocknum} \\
 &= \frac{(p - 1)(p - 2) + (p - 1)(p - 2) + p - 1}{(p - 1)(p - 1)} \\
 &= \frac{2p - 3}{p - 1} = 2 - \frac{1}{p - 1}
 \end{aligned}
 \tag{1}$$

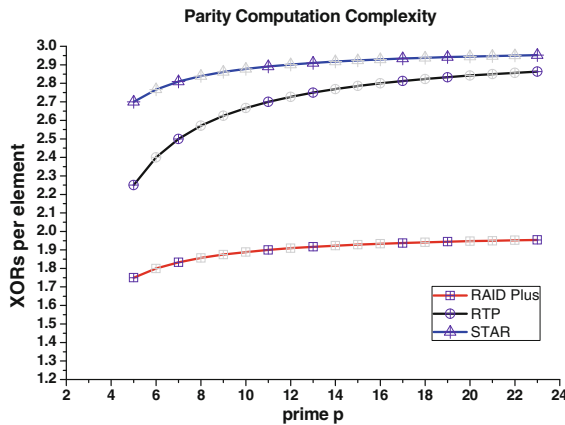


Fig. 4. Parity computation complexity. Obviously, RAID-6Plus uses least XORs per data block for parity computation and is fastest. The difference is more obvious when disk array size is smaller, which is the typical case of most storage systems.

Likely, we can get that of RTP is $A_{RTP} = 3 - \frac{3}{p-1}$. For $(p - 1)p$ data matrix of STAR, similar computation pattern can be applied to get $A_{STAR} = 3 - \frac{1}{p-1}$. The results are shown in Fig. 4.

4.2 Update Penalty

Traditionally, redundancy incurs update penalty because parity must be consistent with data element involved. And update penalty is measured by number of introduced write. For example in RDP, any update on a single data block will result in two more writes to respectively update corresponding row parity and diagonal parity. We can get the update penalty of RTP in reference [2] $U_{RTP} = 5 - \frac{2}{p-1}$ and STAR is $U_{STAR} = 5 - \frac{4}{p}$ with the help of reference [21].

For RAID-6Plus, update penalty of all data elements requires

$$\begin{aligned}
 & [(p - 1)(p - 1) - 2(p - 1) - (p - 2)] \times 2 + 2 \times (p - 1) \times 3 \\
 & + (p - 2) \times 1 = 2p^2 - 3p + 2
 \end{aligned}
 \tag{2}$$

Thus, the update penalty per block is

$$U_{RAID-P} = (2p^2 - 3p + 2)/(p - 1)^2 = 2 + \frac{1}{p - 1} + \frac{1}{(p - 1)^2}
 \tag{3}$$

which is smaller than RTP and STAR, as shown in Fig. 5.

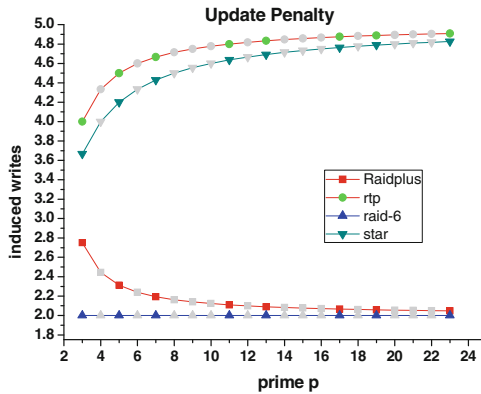


Fig. 5. Update penalty. When p is larger than five, RAID-6Plus introduces least update penalty than RTP and STAR, and it is close to RS-coded RAID6. For example, when p equals 11, RAID-6Plus has 2.11 update penalty compared with 2.82 of RTP and 4.636 of STAR. The different will be more obvious when disk array size grows bigger.

Significantly, RAID-6Plus bears least penalty when updating. The reason why RAID-6Plus has the least update penalty is RAID-6Plus employs short combinations while RTP and STAR use full combinations which involve more data elements to construct, therefore incurring more update penalty.

4.3 Reconstruction Cost

We hereby focus on single failure reconstruction scenario and use total data elements read to evaluate the cost.

For RTP and RDP share exactly same reconstruction sequences, they have the same reconstruction cost, which are respectively $(p - 1)(p - 1)$ with conventional recovery and $\frac{3}{4}(p - 1)^2$ with optimal recovery. Similarly, STAR rebuilds a failed drive with the same cost of EVENODD, which is $(p - 1)p$ conventionally and $\frac{(3p+1)(p-1)}{4}$ by optimal recovery.

When a drive fails in RAID-6Plus, two elements are recovered by two X parities and half of the rest elements are respectively rebuilt with P and Q parity as shown in Fig. 3. Therefore, we compute data elements read for rebuild in RAID-6Plus as

$$\begin{aligned}
 & 2 \times 2 + \frac{(p - 1) - 2}{2} \times (p - 1) + \frac{(p - 1) - 2}{2} \times (p - 1) \\
 & - \frac{(p - 1) - 2}{2} \times \frac{(p - 1) - 2}{2} - 2 = 2 + \frac{(p - 3)(3p - 1)}{4}.
 \end{aligned} \tag{4}$$

We normalize the results based on STAR_conventional in Fig. 6. The secret of RAID-6Plus for single erasure rebuild is it uses short combinations in X drive to minimize elements needed and meanwhile reuse overlapping elements as much as possible on the basis of optimal reconstruction sequence of RDP.

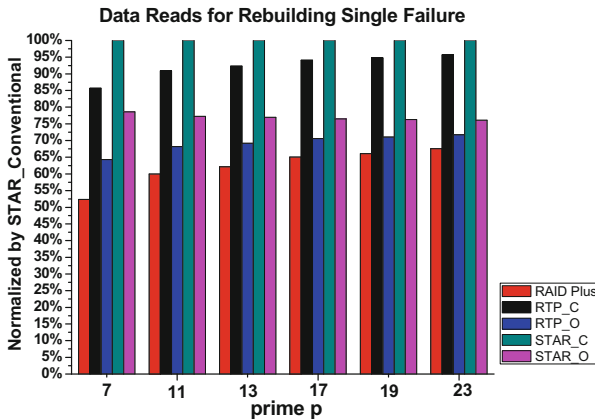


Fig. 6. Data reads for rebuilding single failure. Clearly, RAID-6Plus reads least data than any other coding schemes. For example, when p is 7, RAID-6Plus has a normalized speedup of 33.4%, 11.9%, 47.7% and 26.2% respectively over RTP_Conventional, RTP_Optimal, STAR_Conventional, STAR_Optimal.

4.4 Storage Overhead and Reliability

RTP, STAR are both MDS codes while RAID-6Plus is non-MDS. Regarding storage overheads, all the three codes have three drives dedicated to parities. In terms of absolute failures, RTP and STAR are strictly triple-failure tolerant while RAID-6Plus can tolerate only double failures at device level. But RAID-6Plus provides flexible and higher relative reliability by shorter single failure reconstruction. This effect are hard to be explicitly characterized because there are not any plausible reliability model for RAID systems.

5 Conclusions

Existing triple-failure-tolerant codes set the third parity drive to only handle the triple-failure scenario, which is very rare while single failure dominates in storage system breakdowns in real-world applications.

This study developed RAID-6Plus, a fast and reliable reconstructable coding scheme with shorter reconstruction window. RAID-6Plus remakes the third parity drive to shorten the reconstruction window to degrade multi-failures and to minimize short user wait. The design enables reuse of overlapped elements during reconstruction to balance the reliability and performance of the resulted coding scheme. The performance evaluation indicated that RAID-6Plus exhibited (1) a better system performance with no extra cost comparing to RTP and STAR and (2) an enhanced reliability comparing to RAID-6. The scheme held the potentials of practical uses in modern disk systems, flash-based systems, and even hybrid storage system on any array codes.

Acknowledgment. We are grateful to our anonymous reviewers for their suggestions to improve this paper. This work is supported by the National Natural Science Foundation of China under Grant Nos. 61232003, 61332003, 61202121, 61402503, 61303073.

References

1. Huang, C., Xu, L.: STAR: an efficient coding scheme for correcting triple storage node failures. *IEEE Trans. Comput.* **57**, 889–901 (2008)
2. Goel, A., Corbett, P.: RAID triple parity. *ACM SIGOPS Oper. Syst. Rev.* **46**, 41–49 (2012)
3. Blaum, M., Bruck, J., Vardy, A.: MDS array codes with independent parity symbols. *IEEE Trans. Inf. Theor.* **42**, 529–542 (1996)
4. Jain, N., Dahlin, M., Tewari, R.: TAPER: tiered approach for eliminating redundancy in replica synchronization. In: *FAST*, pp. 21–21
5. Chen, P.M., Lee, E.K., Gibson, G.A., Katz, R.H., Patterson, D.A.: RAID: high-performance, reliable secondary storage. *ACM Comput. Surv. (CSUR)* **26**, 145–185 (1994)
6. Amer, A., Long, D.D., Thomas Schwarz, S.: Reliability challenges for storing exabytes. In: *2014 International Conference on Computing, Networking and Communications (ICNC)*, pp. 907–913. *IEEE* (2014)
7. Schroeder, B., Gibson, G.A.: Disk failures in the real world: what does an MTTF of 1, 000, 000 hours mean to you? In: *FAST*, pp. 1–16
8. Plank, J.S., Blaum, M.: Sector-disk (SD) erasure codes for mixed failure modes in RAID systems. *ACM Trans. Storage (TOS)* **10**, 4 (2014)

9. Leventhal, A.: Triple-parity RAID and beyond. *Queue* **7**, 30 (2009)
10. Xiang, L., Xu, Y., Lui, J., Chang, Q.: Optimal recovery of single disk failure in RDP code storage systems. *ACM SIGMETRICS Perform. Eval. Rev.* **38**, 119–130 (2010)
11. Xiang, L., Xu, Y., Lui, J., Chang, Q., Pan, Y., Li, R.: A hybrid approach to failed disk recovery using RAID-6 codes: algorithms and performance evaluation. *ACM Trans. Storage (TOS)* **7**, 11 (2011)
12. Zhu, Y., Lee, P.P., Xiang, L., Xu, Y., Gao, L.: A cost-based heterogeneous recovery scheme for distributed storage systems with RAID-6 codes, pp. 1–12. *IEEE*
13. Khan, O., Burns, R.C., Plank, J.S., Pierce, W., Huang, C.: Rethinking erasure codes for cloud file systems: minimizing I/O for recovery and degraded reads, p. 20
14. Ma, A., Douglis, F., Lu, G., Sawyer, D., Chandra, S., Hsu, W.: RAIDShield: characterizing, monitoring, and proactively protecting against disk failures. In: *Proceedings of the 13th USENIX Conference on File and Storage Technologies*, pp. 241–256. *USENIX Association* (2015)
15. Mingyuan, X., Mohit, S., Mario, B., David, A.P.: A tale of two erasure codes in HDFS. In: *FAST*, pp. 213–226 (2015)
16. Pinheiro, E., Weber, W.-D., Barroso, L.A.: Failure trends in a large disk drive population. In: *FAST*, pp. 17–23
17. Luo, X., Shu, J.: Load-balanced recovery schemes for single-disk failure in storage systems with any erasure code. In: *2013 42nd International Conference on Parallel Processing (ICPP)*, pp. 552–561. *IEEE* (2013)
18. Boboila, S., Desnoyers, P.: Write endurance in flash drives: measurements and analysis, pp. 9–9
19. Elerath, J.G., Schindler, J.: Beyond MTDL: a closed-form RAID 6 reliability equation. *ACM Trans. Storage (TOS)* **10**, 7 (2014)
20. Corbett, P., English, B., Goel, A., Gracanac, T., Kleiman, S., Leong, J., Sankar, S.: Row-diagonal parity for double disk failure correction. In: *Proceedings of the 3rd USENIX Conference on File and Storage Technologies*, pp. 1–14
21. Rongdong, H., Guangming, L., Jingfei, J.: An efficient coding scheme for tolerating double disk failures. In: *2010 12th IEEE International Conference on High Performance Computing and Communications (HPCC)*, pp. 707–712 (2010)