

Research Report

A New Intra-Disk Redundancy Scheme for High-Reliability RAID Storage Systems in the Presence of Unrecoverable Errors

A. Dholakia*, E. Eleftheriou*, X.-Y. Hu*, I. Iliadis*, J. Menon+, K.K. Rao+

*IBM Research GmbH
Zurich Research Laboratory
8803 Rüschlikon
Switzerland

+IBM Research
Almaden Research Center
650 Harry Road
San Jose, CA
USA

LIMITED DISTRIBUTION NOTICE

This report has been submitted for publication outside of IBM and will probably be copyrighted if accepted for publication. It has been issued as a Research Report for early dissemination of its contents. In view of the transfer of copyright to the outside publisher, its distribution outside of IBM prior to publication should be limited to peer communications and specific requests. After outside publication, requests should be filled only by reprints or legally obtained copies (e.g., payment of royalties). Some reports are available at <http://domino.watson.ibm.com/library/Cyberdig.nsf/home>.



Research
Almaden • Austin • Beijing • Delhi • Haifa • T.J. Watson • Tokyo • Zurich

A New Intra-Disk Redundancy Scheme for High-Reliability RAID Storage Systems in the Presence of Unrecoverable Errors

AJAY DHOLAKIA

IBM Systems and Technology Group

EVANGELOS ELEFThERIOU, XIAO-YU HU, and ILIAS ILIADIS

IBM Zurich Research Laboratory

JAI MENON

IBM Systems and Technology Group

and

KK RAO

IBM Almaden Research Center

Today's data storage systems are increasingly adopting low-cost disk drives that have higher capacity but lower reliability, leading to more frequent rebuilds and to a higher risk of unrecoverable media errors. We propose an efficient intradisk redundancy scheme to enhance the reliability of RAID systems. This scheme introduces an additional level of redundancy inside each disk, on top of the RAID redundancy across multiple disks. The RAID parity provides protection against disk failures, whereas the proposed scheme aims to protect against media-related unrecoverable errors. In particular, we consider an intradisk redundancy architecture that is based on an interleaved parity-check coding scheme, which incurs only negligible I/O performance degradation. A comparison between this coding scheme and schemes based on traditional Reed-Solomon codes and single-parity-check codes is conducted by analytical means. A new model is developed to capture the effect of correlated unrecoverable sector errors. The probability of an unrecoverable failure associated with these schemes is derived for the new correlated model as well as for the simpler independent error model. We also derive closed-form expressions for the mean time to data loss of RAID 5 and RAID 6 systems in the presence of unrecoverable errors and disk failures. We then combine these results to characterize the reliability of RAID systems that incorporate the intradisk redundancy scheme. Our results show that in the practical case of correlated errors, the interleaved parity check scheme provides the same reliability as the optimum, albeit more complex, Reed-Solomon coding scheme. Finally, the I/O and throughput performance are evaluated by means of analysis and event-driven simulations.

Categories and Subject Descriptors: B.3.2 [Memory Structures]: Design Styles—*mass storage (e.g., magnetic, optical, RAID)*; B.8.1 [Performance and Reliability]: Reliability, Testing, and Fault-Tolerance; C.4 [Computer Systems Organization]: Performance of Systems—*Fault tolerance; Modeling techniques*; H.3.4 [Information Storage and Retrieval]: Systems and Software—*Performance evaluation (efficiency and effectiveness)*

An earlier version of this article was presented at the *Joint International Conference on Measurement and Modeling of Computer Systems (ACM SIGMETRICS/IFIP Performance 2006)* Saint Malo, France.

Authors' addresses: Ajay Dholakia, IBM Systems and Technology Group, 3039 E Cornwallis Road, P.O. Box 12195, Research Triangle Park, NC 27709-2195; Evangelos Eleftheriou, Xiao-Yu Hu, and Ilias Iliadis, IBM Research GmbH, Zurich Research Laboratory, Säumerstrasse 4, 8803 Rüschlikon, Switzerland; Jai Menon, IBM Systems and Technology Group, 5600 Cottle Road, San Jose, CA 95193-0003; KK Rao, IBM Research, Almaden Research Center, 650 Harry Road, San Jose, CA 95120-6099. The contact author is I. Iliadis.

General Terms: Performance, Reliability

Additional Key Words and Phrases: File and I/O systems, RAID, reliability analysis, stochastic modeling

1. INTRODUCTION

Large-capacity data-storage systems are ubiquitous in modern enterprises, and the demand for more capacity continues to grow. Such data storage systems use hundreds of hard-disk drives (HDDs) to achieve the required aggregate data capacity. A problem encountered in these systems is failures of HDDs. Protection against such failures is achieved by employing redundant disks in a system. The common technique used in modern data storage systems for tolerating disk failures is the redundant array of independent disks (RAID) [Chen et al. 1994; Patterson et al. 1988]. A popular RAID scheme is RAID Level 5, in which disks are arranged in groups (or arrays), each with one redundant disk. RAID 5 arrays can tolerate one disk failure per array. In addition, data striping and distributed parity placement across multiple disks are used to benefit from faster parallel access and load balancing.

As the number of disks in a data storage system grows, also the need for tolerating two disk failures in an array increases. The RAID 5 scheme cannot protect against data loss if two disks fail. Instead, using a RAID 6 scheme allows up to two disks to fail in an array. The RAID 6 scheme stores two parity strips (stripe units) per stripe set [Blaum et al. 1995; Corbett et al. 2004]. However, this increase in reliability reduces the overall throughput performance of RAID 6 arrays as well as the available storage space for a fixed number of total disks in an array. The main reason for the reduced throughput is that each write request also requires updating the two corresponding parity units on different disks.

A current trend in the data storage industry is towards the adoption of low-cost components, most notably SATA disk drives instead of FC and SCSI disk drives. SATA drives offer higher capacity per drive, but have a comparatively lower reliability. As the disk capacity grows, the total number of bytes that are read during a rebuild operation becomes very large. This increases the probability of encountering an unrecoverable error, i.e., an error that cannot be corrected by either the standard sector-associated error-control coding (ECC) or the re-read mechanism of the HDD. Unrecoverable media errors typically result in one or more sectors becoming unreadable. This is particularly problematic when combined with disk failures. For example, if a disk fails in a RAID 5 array, the rebuild process must read all the data on the remaining disks to rebuild the lost data on a spare disk. During this phase, a media error on any of the good disks would be unrecoverable and lead to data loss because there is no way to reconstruct the lost data sectors. A similar problem occurs when two disks fail in a RAID 6 scheme. In this case, any unrecoverable sectors encountered on the good disks during the rebuild process also lead to data loss. Typical data storage installations also include a tape-based back-up or a disk-based mirrored copy at a remote location. These mechanisms can be used to reconstruct data lost because of unrecoverable errors. However, there is

a significant penalty in terms of latency and throughput.

We propose a new technique to enhance the reliability of RAID schemes that incurs only a negligible I/O performance degradation and is based on intra-disk redundancy. The method introduces an additional “dimension” of redundancy inside each disk that is orthogonal to the usual RAID dimension based on redundancy across multiple disks. The RAID redundancy provides protection against disk failures, whereas the proposed intra-disk redundancy aims to protect against media-related unrecoverable errors.

The basic intra-disk redundancy scheme works as follows: each strip (stripe unit) is partitioned into segments, and within each segment, a portion of the storage, usually several sectors (called data sectors), is used for storing data, whereas the remainder is reserved for redundant sectors, which are computed based on an erasure code. The novelty of the proposed scheme lies in the fact that it copes with precisely those type of errors that cannot be handled by the built-in ECC and re-read mechanisms of an HDD. It can also be used to address similar “data-integrity” errors such as bit-flips and other incorrect responses. Furthermore, we address the placement issue of the redundant sectors within the segment to minimize the impact on the throughput performance.

The key contributions of this paper are the following. A new intra-disk redundancy scheme for high-reliability RAID storage systems is introduced for erasure correction in the presence of unrecoverable sector errors. In particular, we consider an intra-disk redundancy architecture that is based on a simple XOR-based interleaved parity-check (IPC) coding scheme. Furthermore, a new model capturing the effect of correlated unrecoverable sector errors is developed and subsequently used to analyze the proposed coding scheme as well as the traditional redundancy schemes based on Reed–Solomon (RS) codes and single-parity-check (SPC) codes. The probability of an unrecoverable failure associated with these schemes is derived for the new correlated model as well as for the simpler independent error model. Furthermore, suitable Markov models are developed to derive closed-form expressions for the mean time to data loss (MTTDL) of RAID 5 and RAID 6 systems in the presence of unrecoverable errors and disk failures. We then combine these results to comprehensively characterize the reliability of these RAID systems if complemented with the intra-disk redundancy scheme. Finally, the I/O and throughput performance of these RAID systems is evaluated by means of analysis and by using event-driven simulations under a variety of workloads.

As our results demonstrate, the easy-to-implement IPC coding scheme considered here achieves a reliability very close to that of the optimal but much more complex RS scheme. As will be explored in further detail in this paper, a key advantage of the intra-disk redundancy scheme is that it can be applied to various RAID systems, including RAID 5 and RAID 6. It can also be applied in conjunction with any other mechanism that is used to reduce the number of unrecoverable errors and thereby improve reliability, such as, for example, disk scrubbing. Furthermore, the analytical expressions derived can also be used to obtain the system reliability in this context, based on the adjusted probability of encountering an unrecoverable error.

The remainder of the paper is organized as follows. Section 2 provides a survey of

the relevant literature on reliability-enhancement schemes for RAID systems. Section 3 describes the problem of data loss due to unrecoverable errors in more detail. Section 4 presents the intra-disk redundancy scheme, with the relevant performance measures being considered in Section 5. Section 6 provides a detailed analysis of the erasure correction capability of the various coding schemes in the presence of independent as well as correlated unrecoverable sector errors. In Section 7 closed-form expressions for the reliability of RAID 5 and RAID 6 storage systems that incorporate the intra-disk redundancy scheme are derived. The I/O performance is evaluated analytically in Section 8. Section 9 presents numerical results demonstrating the effectiveness of the proposed scheme. An analytical investigation of the reliability and sensitivity to the various parameters is conducted, and the I/O response time and throughput performance are evaluated by means of simulation. Finally, we conclude in Section 10.

2. RELATED WORK

Data storage systems are designed to meet increasingly more stringent data-integrity requirements [Keeton et al. 2004]. Using a tape-based back-up or a disk-based mirrored copy is the approach commonly used to enhance data integrity. However, recovering data from such copies is time consuming.

The emergence of SATA drives as low-cost alternative to SCSI and FC drives in data storage systems has brought the issue of system reliability to the forefront. The key problem with SATA drives in this respect is that unrecoverable errors are ten times more likely than on SCSI/FC drives [Hitachi Global Storage Technologies 2007]. A simple scheme based on using intra-disk redundancy is described in [Hughes and Murray 2004] and aims at increasing the reliability of SATA drives to the same level as that of SCSI/FC drives. This scheme is based on using a single parity sector for a large number of data sectors, but does not address its placement issue. In the case of small writes, the data and parity sectors to be updated will require separate I/O requests, leading to a severe penalty in throughput performance.

Following the introduction of RAID [Patterson et al. 1988], the reliability of RAID systems was analyzed by several groups. A basic reliability analysis of RAID systems was presented in [Burkhard and Menon 1993; Malhotra and Trivedi 1993; Schulze et al. 1989]. Unrecoverable errors were considered in [Malhotra and Trivedi 1995], where a detailed Markov model is developed to capture a variety of failures possible in a disk array. The model also incorporated uncorrectable permanent errors caused by media-related errors. In [Wu et al. 1997], the reliability of RAID 5 arrays in the presence of uncorrectable bit errors was analyzed. The authors assume that reading data from disks does not cause uncorrectable errors. These errors are assumed to occur during writing and are then encountered during reading. A separate analysis of two cases is done: one in which uncorrectable errors exist on good disks before a disk failure, the other in which uncorrectable errors occur during writes to good disks after a disk failure but before the rebuild is completed. The latter scenario captures the case when the disk array continues to receive read and write requests during the rebuild phase. The authors use Markov models to characterize the occurrence of uncorrectable errors and obtain expressions for the

reliability of RAID 5 arrays. They demonstrate that unrecoverable errors have a big impact on the reliability of the system.

More recently, the reliability of large storage systems that encounter disk failures as well as unrecoverable errors was evaluated in [Xin et al. 2003]. The use of a signature scheme was proposed to identify unrecoverable blocks. Redundancy was introduced based on two-way mirroring, three-way mirroring, and RAID 5 with mirroring (RAID 5+1). The redundancy in the schemes analyzed was placed on different disks to protect against disk failures, thus exploiting the RAID dimension. The reliability of these schemes was analyzed using Markov models. In [Chen and Towsley 1996], the performance of different RAID systems was studied and various scheduling policies were presented. More recently, an integrated performance model was developed in [Varki et al. 2004] that incorporated several features of real disk arrays such as caching, parallelism and array controller optimizations.

3. DATA LOSS FROM UNRECOVERABLE ERRORS

In this section, we consider the problem of unrecoverable errors and their impact on the reliability of a RAID 5 system to motivate the need for devising a coding scheme.

Consider an example of a number of RAID 5 systems installed in the field. Each system may contain more than one RAID 5 array. What is important is the total number of RAID 5 arrays. We assume that all arrays have the same parameters. Consider an installed base of $n_G = 125000$ RAID 5 arrays, each with $N = 8$ disks. All the systems in the field are assumed to comprise the same type of disks. Two types of disks are assumed, either the expensive and highly reliable SCSI drives or the low-cost SATA drives with lower reliability. The disks are characterized by the following parameters:

- Drive capacity (C_d): SCSI drives with 73, 146, and 300 GB, and SATA drives with 300 and 500 GB.
- Mean time to failure ($1/\lambda$): 1×10^6 h for SCSI and 5×10^5 h for SATA drives.
- Mean time to rebuild ($1/\mu$): 9.3 h for 146 GB SCSI and 17.8 h for 300 GB SATA drives.
- Unrecoverable bit error probability (P_{bit}): 1×10^{-15} for SCSI and 1×10^{-14} for SATA drives.

Assuming a sector size of 512 bytes, the equivalent unrecoverable sector error probability is $P_s \approx P_{\text{bit}} \times 4096$, which is 4.096×10^{-12} in the case of SCSI and 4.096×10^{-11} in the case of SATA drives.

For a RAID 5 array, the unrecoverable errors lead to data loss when encountered in the critical mode, i.e. when one drive has already failed. In this case, the remaining $N - 1$ drives are read to rebuild the data of the failed drive. As the number of sectors on a drive is $C_d/512$, the total number of sectors read while rebuilding from $N - 1$ drives is $(N - 1)C_d/512$. Assuming each sector encounters an unrecoverable error independently of all other sectors with probability P_s , the probability of encountering at least one unrecoverable sector, i.e., the probability of an unrecoverable failure P_{uf} is given by

$$P_{\text{uf}} = 1 - (1 - P_s)^{(N-1)C_d/512} . \quad (1)$$

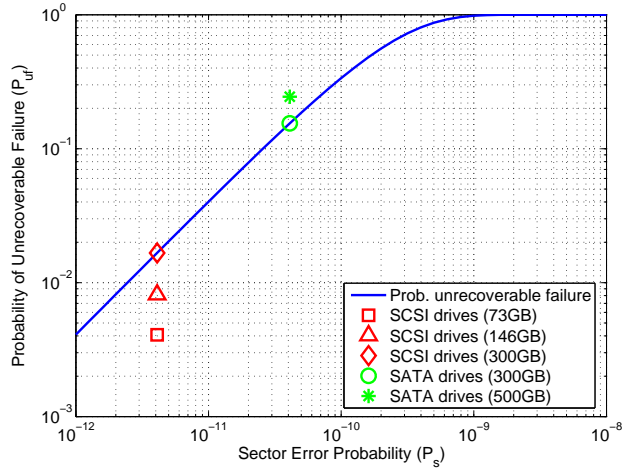


Fig. 1. P_{uf} as a function of P_s .

Figure 1 shows P_{uf} for disks of 300 GB capacity as a function of the unrecoverable sector error probability. Also shown are the results for SCSI drives with three different capacities and SATA drives with two different capacities. An array with 300 GB SCSI drives has a P_{uf} of more than 1%. For arrays using the low-cost SATA drives with 500 GB capacity, P_{uf} increases to more than 25%.

The detrimental effect of unrecoverable failures on the overall data loss experienced by users of large storage systems can be seen by examining the MTTDL metric. In the presence of disk failures only, the MTTDL of a RAID 5 array system is well known [Chen et al. 1994; Patterson et al. 1988] and is given by

$$MTTDL = \frac{\mu}{n_G N(N-1)\lambda^2}, \quad (2)$$

assuming $\lambda \ll \mu$. The MTTDL of a large data storage installation of RAID 5 arrays as a function of the total user capacity is shown in Figure 2. It can be seen that a 10 PB installation using either SCSI or SATA drives has an MTTDL of more than five years. Bringing unrecoverable failures into consideration changes the picture dramatically. Using the expression for the MTTDL in the presence of both disk failures and unrecoverable failures, which we derive in Section 7.2, a 10 PB installation using 146 GB SCSI drives experiences a MTTDL of approximately ten weeks, as shown in Figure 2. More interestingly, the MTTDL of a 10 PB installation using 300 GB SATA drives drops to less than one week. These examples clearly show that data loss resulting from unrecoverable sectors is a key limitation of current large-scale data storage systems.

4. INTRA-DISK REDUNDANCY SCHEME

Here we introduce and describe the intra-disk redundancy scheme. A number of contiguous data sectors in a strip as well as redundant sectors derived from these data sectors are grouped together forming a segment. A number of different schemes can be used to obtain the redundant parity sectors, as will be described later in

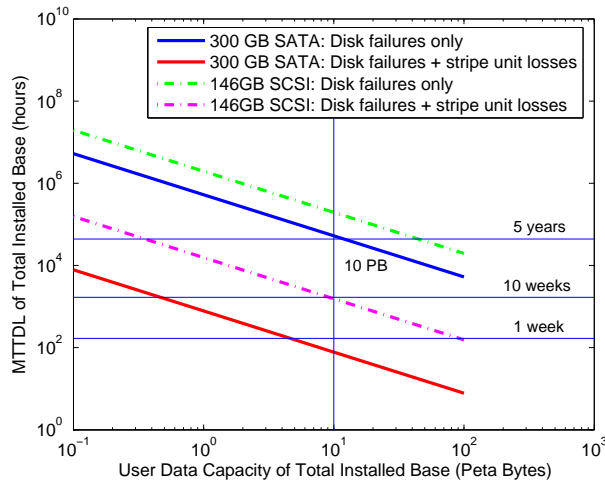


Fig. 2. MTTDL of RAID 5 arrays as a function of total installed user capacity.

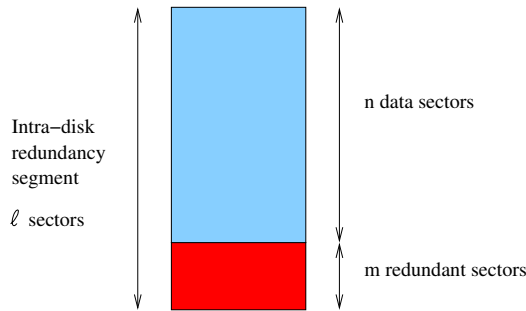


Fig. 3. Basic intra-disk redundancy scheme.

this section. The entire segment, comprising ℓ data and parity sectors, is stored contiguously on the same disk, as shown in Figure 3, where $\ell = n + m$.

The size of a segment should be chosen such that sufficient degrees of storage efficiency, performance and reliability are ensured. For practical reasons, the strip size should be a multiple of the data-segment size. In addition, the number m of parity sectors in a segment is a design parameter that can be optimized based on the desired set of operating conditions. In general, more redundancy (large m) provides more protection against unrecoverable media errors. However, it also incurs more overhead in terms of storage space and computations required to obtain and update the parity sectors. Furthermore, for a fixed degree of storage efficiency, increasing the segment size results in an increased reliability, but also in an increased penalty on the I/O performance. Therefore, a judicious trade-off between these competing requirements needs to be made. The storage efficiency $se^{(\text{IDR})}$ of the intra-disk

Table I. Notation of System Parameters

Notation	Parameter
N	Number of disks per array group
n_G	Number of array groups in the system
C_d	Disk drive capacity
S	Sector size
ℓ	Number of sectors in a segment
n_d	Number of segments in a disk drive
m	Number of parity sectors in a segment, number of interleaves, or interleaving depth
$1/\lambda$	Mean time to failure for a disk
P_{bit}	Probability of an unrecoverable bit error
$se^{(\text{IDR})}$	Storage efficiency of the intra-disk redundancy scheme
$se^{(\text{RAID})}$	Storage efficiency of the RAID scheme
$se^{(\text{RAID+IDR})}$	Overall storage efficiency of the entire system
$1/\mu$	Mean time to rebuild in the critical mode for a RAID 5 array
$1/\mu_1$	Mean time to rebuild in the degraded mode for a RAID 6 array
$1/\mu_2$	Mean time to rebuild in the critical mode for a RAID 6 array
P_s	Probability of an unrecoverable sector error
P_{seg}	Probability of a segment encountering an unrecoverable sector error
P_{uf}	Probability of an unrecoverable failure

redundancy scheme is given by

$$se^{(\text{IDR})} = \frac{\ell - m}{\ell}. \quad (3)$$

5. SYSTEM ANALYSIS

The notation used for the purpose of our analysis is given in Table I. The parameters are divided into two sets, namely, the set of independent and that of dependent parameters listed in the upper and lower part of the table, respectively.

Assuming that errors occur independently over successive bits, the unrecoverable sector error probability P_s is given by

$$P_s = 1 - (1 - P_{\text{bit}})^S, \quad (4)$$

with S expressed in bits.

Similarly, when no coding within the segment is applied ($m = 0$), the unrecoverable segment error probability P_{seg} is given by

$$P_{\text{seg}}^{\text{None}} = 1 - (1 - P_s)^\ell = 1 - (1 - P_{\text{bit}})^{S\ell}, \quad (5)$$

with S expressed in bits.

As the segment size is equal to ℓS , the number of segments in a disk drive, n_d , is given by

$$n_d = \frac{C_d}{\ell S}. \quad (6)$$

A RAID array is considered to be in the *critical mode* when an additional disk failure can no longer be tolerated. Thus, RAID 5 and RAID 6 arrays are in critical mode when they operate with one disk and two disks failed, respectively. Also, a

RAID 6 array is considered to be in the *degraded mode* when it operates with one disk failed. An unrecoverable failure occurs when an array is in the critical mode and at least one of the n_s segments that need to be read is in error. Consequently, the probability of an unrecoverable failure, P_{uf} , is given by

$$P_{\text{uf}} = 1 - (1 - P_{\text{seg}})^{n_s} . \quad (7)$$

For a RAID 5 and a RAID 6 system in the critical mode, the corresponding probabilities of an unrecoverable failure $P_{\text{uf}}^{(1)}$ and $P_{\text{uf}}^{(2)}$ are obtained by setting $n_s = (N - 1)n_d$ and $n_s = (N - 2)n_d$, as there are $N - 1$ and $N - 2$ operational disks, respectively. From (6), it follows that

$$P_{\text{uf}}^{(1)} = 1 - (1 - P_{\text{seg}})^{\frac{(N-1)C_d}{\ell S}} , \quad (8)$$

and

$$P_{\text{uf}}^{(2)} = 1 - (1 - P_{\text{seg}})^{\frac{(N-2)C_d}{\ell S}} . \quad (9)$$

The probability P_{seg} of the various coding schemes is evaluated in Section 6.

5.1 Storage Efficiency

The storage efficiency of the RAID scheme chosen is given by

$$se^{(\text{RAID})} = \frac{N - p}{N} , \quad (10)$$

with

$$p = \begin{cases} 1 & \text{for a RAID 5 system} \\ 2 & \text{for a RAID 6 system.} \end{cases} \quad (11)$$

Note that the above expressions hold for a scheme not using intra-disk redundancy. If an intra-disk redundancy scheme is used, the overall storage efficiency of the entire array (or system) is given by

$$se^{(\text{RAID+IDR})} = se^{(\text{RAID})} se^{(\text{IDR})} = \left(1 - \frac{p}{N}\right) \left(1 - \frac{m}{\ell}\right) . \quad (12)$$

5.2 Rebuild Time

The time required to rebuild depends on the various parameters listed in Table II, which include the drive capacity and the bandwidth that the drive provides. The first part of the table lists the independent parameters.

We assume that during a disk rebuild, the disk array continues to actively service I/O requests, which implies that every rebuild command requires a seek. The utilization factor U refers to the fraction of time the controller spends performing rebuilds as opposed to servicing I/O requests. The size s_{req} of the read/write requests issued to the drive is usually 64 to 256 KB. The average read/write operations per second, r_{io} , for single randomly chosen sectors are around 250 to 300 for SCSI, and 150 for SATA. This term accounts for seeks and rotational latency. The average disk transfer rate t_d following the seek and rotational latency is typically 60-100 MB/s for SCSI and 40-60 MB/s for SATA. The sustained bandwidth b_m of the memory subsystem is typically of the order of GB/s.

Table II. System Parameters Affecting Rebuild Time

Notation	Parameter
U	Utilization factor
s_{req}	Size of read/write requests issued to the drive
r_{io}	Rate of random single-sector disk read/write operations
t_d	Average disk transfer rate
b_m	Sustained bandwidth of memory subsystem
C_d	Disk drive capacity
N	Number of disks per array group
b_d	Effective disk-rebuild bandwidth

From the above, it follows that the average time required by an I/O request to complete is equal to the sum of $1/r_{\text{io}}$, the seek and rotational latency, and s_{req}/t_d , the transfer time. Therefore, the effective disk-rebuild bandwidth b_d is given by

$$b_d = \frac{s_{\text{req}}}{\frac{1}{r_{\text{io}}} + \frac{s_{\text{req}}}{t_d}}. \quad (13)$$

The mean time to rebuild, assuming that the bottleneck is encountered at the disk drives, is then given by $C_d/(b_d U)$. We now proceed with the evaluation of the mean time to rebuild, assuming that the bottleneck is encountered at the memory subsystem.

Let us first consider a RAID 5 array in the degraded mode. During a rebuild of a data unit, $N - 1$ data units are being read in parallel from the disks and transferred to the memory, and then from the memory to the XOR engine. The XOR operation yields a data unit, which is written back to the memory and then from the memory to the disk. Consequently, the total number of data units transferred through the memory subsystem is equal to $2(N - 1) + 2 = 2N$. Assuming approximately equal read and write speed, the time required for a disk rebuild is equal to $C_d 2N/(b_m U)$. It now follows that the mean time to rebuild is the maximum of the two times evaluated, i.e.

$$\mu^{-1} = \max\left(\frac{C_d}{b_d U}, \frac{2NC_d}{b_m U}\right) = \frac{C_d}{U} \max\left(\frac{1}{b_d}, \frac{2N}{b_m}\right). \quad (14)$$

Let us now consider a RAID 6 array in the degraded mode. As the rebuild of a data unit is performed based on $N - 2$ data units, the total number of data units transferred through the memory subsystem is equal to $2(N - 2) + 2 = 2(N - 1)$. Consequently, the mean time to rebuild is given by

$$\mu_1^{-1} = \max\left(\frac{C_d}{b_d U}, \frac{2(N - 1)C_d}{b_m U}\right) = \frac{C_d}{U} \max\left(\frac{1}{b_d}, \frac{2(N - 1)}{b_m}\right). \quad (15)$$

Finally, let us now consider a RAID 6 array in the critical mode. To rebuild the two strips of a stripe that correspond to the two failed disks, $N - 2$ strips are being read in parallel from the disks and transferred to the memory. They are subsequently transferred from the memory to the XOR engine twice in order to

Table III. Numerical values and results

Parameter	Value	
	SCSI	SATA
U	0.10	0.20
s_{req}	256 MB	
r_{io}	300 op/s	150 op/s
t_d	100 MB/s	60 MB/s
b_m	2 GB/s	
C_d	146 GB	300 GB
N	8 for RAID 5 16 for RAID 6	
b_d	43.4 GB/s	23.4 GB/s
$b_m/(2N)$	125 GB/s for RAID 5	
$b_m/[2(N-1)]$	66.6 GB/s for RAID 6	
$b_m/(3N-2)$	43.5 GB/s for RAID 6	
$1/\mu$	9.3 h	17.8 h
$1/\mu_1$	9.3 h	17.8 h
$1/\mu_2$	9.3 h	17.8 h

perform the two XOR operations required for retrieving the two lost strips. The two resulting strips are written back to the memory and then from the memory to the disk. Consequently, the total number of data units transferred through the memory subsystem is equal to $3(N-2) + 4 = 3N - 2$. Note that all working drives are being read in parallel, whereas the two drives that are being rebuilt are being written in parallel. Hence, they are all equally bottlenecked (assuming approximately equal read and write speeds). Consequently, the mean time to rebuild is given by

$$\mu_2^{-1} = \max\left(\frac{C_d}{b_d U}, \frac{(3N-2)C_d}{b_m U}\right) = \frac{C_d}{U} \max\left(\frac{1}{b_d}, \frac{3N-2}{b_m}\right). \quad (16)$$

The parameter values assumed and the corresponding results obtained are listed in the upper and lower part of Table III, respectively. The utilization factor is considered to be higher in the case of SATA drives because the I/O activity of a system using SATA drives is likely to be less than that of a system using SCSI drives. Note that in all cases the bottleneck during a rebuild operation is encountered at the disk drives.

6. INDEPENDENT AND CORRELATED ERRORS

The performance of the intra-disk redundancy scheme is analytically assessed based on two models. According to the first model (independent model), each sector encounters an unrecoverable error, independently of all other sectors, with probability P_s . This implies that the lengths (in number of sectors) of error-free intervals are independent and geometrically distributed with parameter P_s . In addition, we introduce a model for capturing error-correlation effects in which sector errors are assumed to occur in bursts. We refer to this model as the correlated model. Let B and I denote the lengths (in number of sectors) of bursts and of the error-free intervals between successive bursts, respectively. Let \bar{B} and \bar{I} denote the corresponding average lengths. These lengths are assumed to be i.i.d., i.e. independently and

identically distributed random variables. In particular, the error-free intervals are assumed to be geometrically distributed, as in the independent model, but with a parameter α . Therefore, the probability density function (pdf) $\{a_j\}$ of the length j of a typical error-free interval is given by $a_j = P(I = j) = \alpha(1 - \alpha)^{j-1}$ for $j = 1, 2, \dots$, such that $\bar{I} = 1/\alpha$, with $0 < \alpha \leq 1$. Let also $\{b_j\}$ denote the pdf of the length j of a typical burst of consecutive errors, i.e. $P(B = j) = b_j$ for $j = 1, 2, \dots$. The average burst length is then given by $\bar{B} = \sum_{j=1}^{\infty} j b_j$ and is assumed to be bounded. Owing to ergodicity, the probability P_s that an arbitrary sector has an unrecoverable error is given by

$$P_s = \frac{\bar{B}}{\bar{B} + \bar{I}}. \quad (17)$$

From the above, it follows that

$$\alpha = \frac{P_s}{B(1 - P_s)} = \frac{P_s}{B} + \frac{P_s^2}{B} + \frac{P_s^3}{B} + \dots = \frac{P_s}{B} + O(P_s^2) \approx \frac{P_s}{B}, \quad (18)$$

and that

$$P_s \leq \frac{\bar{B}}{\bar{B} + 1}, \quad (19)$$

given that $\alpha \leq 1$, or, equivalently, $\bar{I} \geq 1$. This approximation as well as the ones derived below are valid when P_s is quite small, in which case terms involving powers of P_s to higher orders are negligible and can be ignored.

Note that the independent model is a special case of the correlated model in which the $\{b_j\}$ distribution is geometric with parameter $1 - P_s$, i.e. $b_j = (1 - P_s)P_s^{j-1}$ for $j = 1, 2, \dots$, and $\bar{B} = 1/(1 - P_s)$. Let $\{G_n\}$ denote the complementary cumulative density function (ccdf) of the burst length B . Then G_n denotes the probability that the length of a burst is greater than or equal to n , i.e. $G_n \triangleq \sum_{j=n}^{\infty} b_j$, for $n = 1, 2, \dots$. In this case, and for a given m ($m \in \mathbb{N}$), it holds that the probability G_{m+1} that a burst of more than m consecutive errors occurs is negligible because $G_{m+1} = P_s^m \ll P_s$. In the remainder of the paper, however, we consider fixed (independent of P_s) burst distributions for which this probability is nonnegligible, i.e. $G_{m+1} \gg P_s$, and the average burst length is relatively small, i.e. $\bar{B} \ll 1/P_s$. Consequently, the results for the independent model need to be obtained separately as they cannot be derived from those for the correlated model.

Let us consider the sectors divided into groups of ℓ ($\ell > m$) successive sectors, with each such group constituting a segment. If no coding scheme is applied ($m = 0$), a segment is in error if there is an unrecoverable sector error. For the independent model, the probability P_{seg} that a segment is in error is then given by

$$P_{\text{seg}}^{\text{None}} = 1 - (1 - P_s)^\ell = \ell P_s + O(P_s^2) \approx \ell P_s. \quad (20)$$

For the correlated model, the segment is correct if the first sector is correct and the subsequent $\ell - 1$ sectors are also correct. The probability of the first sector being correct is equal to $1 - P_s$, whereas from the geometric assumption the probability of each subsequent sector being correct is equal to $1 - \alpha$. By making use of (18) we

obtain

$$\begin{aligned} P_{\text{seg}}^{\text{None}} &= 1 - (1 - P_s)(1 - \alpha)^{\ell-1} = 1 - (1 - P_s) \left(1 - \frac{P_s}{B}\right)^{\ell-1} = \\ &= \left(1 + \frac{\ell-1}{B}\right) P_s + O(P_s^2) \approx \left(1 + \frac{\ell-1}{B}\right) P_s. \end{aligned} \quad (21)$$

We now proceed with the evaluation of P_{seg} for various coding schemes. In particular, we consider P_{seg} expressed as a series expansion in powers of P_s , i.e. $P_{\text{seg}} = \sum_{i=1}^{\infty} c_i P_s^i$, with the coefficients c_i being independent of P_s . It turns out that in the case of the correlated model and for small P_s , the performance of the coding schemes considered can be obtained by considering the power series taken to the first order. That is, it suffices to make a power series expansion of P_{seg} in P_s of the form $P_{\text{seg}} = c_1 P_s + O(P_s^2)$. First we establish the following propositions which hold for the correlated model and independently of the coding scheme used.

PROPOSITION 6.1. *The probability $P_{\text{seg}}^{(k)}$ that a segment contains k ($k \leq \ell/2$) bursts of errors and is in error is of order $O(P_s^k)$.*

PROOF. See Appendix A. \square

PROPOSITION 6.2. *It holds that $P_{\text{seg}} = c_1 P_s + O(P_s^2)$, with c_1 derived based only on $P(\text{segment contains a single burst of errors and is in error})$.*

PROOF. By conditioning on the number of bursts of errors in a segment and using Proposition 6.1, we obtain

$$\begin{aligned} P_{\text{seg}} &= \sum_{k=1}^{\ell/2} P(\text{segment contains } k \text{ bursts of errors and is in error}) = \sum_{k=1}^{\ell/2} P_{\text{seg}}^{(k)} = \\ &= P(\text{segment contains a single burst of errors and is in error}) + \sum_{k=2}^{\ell/2} P_{\text{seg}}^{(k)} = \\ &= P(\text{segment contains a single burst of errors and is in error}) + O(P_s^2). \end{aligned} \quad (22)$$

\square

6.1 Reed–Solomon (RS) Coding

Reed–Solomon (RS) coding is the standard choice for erasure correction when implementation complexity is not a constraint. This is because these codes provide the best possible erasure correction capability for a given number of parity symbols, i.e. for a given storage efficiency (code rate). Essentially, for a code with m parity symbols in a codeword of n symbols, any m erasures in the block of n symbols can be corrected. RS codes are used in a wide variety of applications and are the primary mechanism that allows the stringent uncorrectable error probability specification of HDDs to be met. Note that the RS codes considered here provide an additional level of redundancy to that of the built-in ECC scheme.

The performance of the RS scheme is the best that can be achieved. With such a code, the probability of a segment being in error is equal to the probability of

getting more than m unrecoverable sector errors per segment and is given by

$$\begin{aligned} P_{\text{seg}}^{\text{RS}} &= \sum_{j=m+1}^{\ell} \binom{\ell}{j} P_s^j (1 - P_s)^{\ell-j} = \binom{\ell}{m+1} P_s^{m+1} + O(P_s^{m+2}) \approx \\ &\approx \binom{\ell}{m+1} P_s^{m+1}. \end{aligned} \quad (23)$$

In the case of the correlated model, an approximate expression for the probability of a segment being in error is given by the following theorem.

THEOREM 6.3. *It holds that*

$$P_{\text{seg}}^{\text{RS}} = c_1^{\text{RS}} P_s + O(P_s^2), \quad (24)$$

where

$$c_1^{\text{RS}} = 1 + \frac{(\ell - m - 1)G_{m+1} - \sum_{j=1}^m G_j}{\bar{B}}. \quad (25)$$

PROOF. See Appendix B. \square

COROLLARY 6.4. *For small values of P_s , it holds that*

$$P_{\text{seg}}^{\text{RS}} \approx \left[1 + \frac{(\ell - m - 1)G_{m+1} - \sum_{j=1}^m G_j}{\bar{B}} \right] P_s. \quad (26)$$

COROLLARY 6.5. *The coefficient c_1^{RS} is equal to zero if and only if $G_{m+1} = 0$, i.e. the maximum burst length does not exceed m .*

PROOF. Note that c_1^{RS} can also be written as $[(\ell - m)G_{m+1} + \sum_{j=m+2}^{\infty} G_j]/\bar{B}$, which is equal to zero if and only if $G_{m+1} = 0$. \square

6.2 Single-Parity Check (SPC) Coding

The simplest coding scheme is one in which a single parity sector is computed by using the XOR operation on $\ell - 1$ data sectors to form a segment with ℓ sectors in total. Such a scheme can tolerate a single erasure anywhere in the segment. In fact, the parity in a RAID 5 scheme is based on such a single parity-check (SPC) scheme, albeit with the redundancy along the RAID dimension. The probability of a segment being in error is equal to the probability of getting at least two unrecoverable sector errors. The independent model yields

$$P_{\text{seg}}^{\text{SPC}} = \sum_{j=2}^{\ell} \binom{\ell}{j} P_s^j (1 - P_s)^{\ell-j} = \frac{\ell(\ell-1)}{2} P_s^2 + O(P_s^3) \approx \frac{\ell(\ell-1)}{2} P_s^2. \quad (27)$$

In the case of the correlated model, an approximate expression for the probability of a segment being in error is given by the following theorem.

THEOREM 6.6. *It holds that*

$$P_{\text{seg}}^{\text{SPC}} = c_1^{\text{SPC}} P_s + O(P_s^2), \quad (28)$$

where

$$c_1^{\text{SPC}} = 1 + \frac{(\ell-2)G_2 - 1}{\bar{B}}. \quad (29)$$

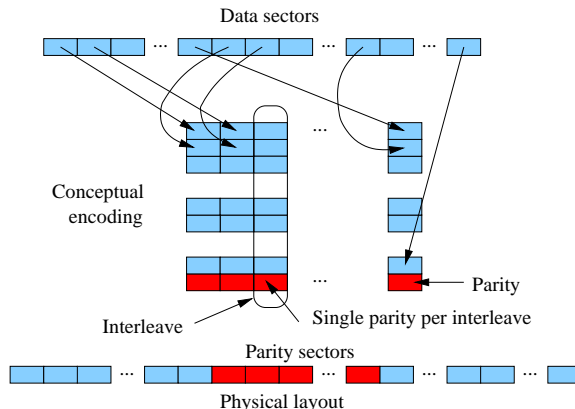


Fig. 4. Intra-disk redundancy scheme using the interleaved parity-check coding scheme.

PROOF. Note that the SPC coding scheme is a special case of the RS coding scheme in which only a single sector error can be corrected in a segment. Expressions (28)–(29) are therefore derived from (24)–(25) by setting $m = 1$. \square

COROLLARY 6.7. For small values of P_s , it holds that

$$P_{\text{seg}}^{\text{SPC}} \approx \left[1 + \frac{(\ell - 2)G_2 - 1}{B} \right] P_s. \quad (30)$$

6.3 Interleaved Parity-Check (IPC) Coding

A coding scheme called interleaved parity check (IPC), which has a simplicity akin to that of the SPC scheme but considerably better performance is introduced next. In this scheme, n ($n = \ell - m$) contiguous data sectors are conceptually arranged in a matrix as shown in Figure 4. Data sectors in a column are XORed to obtain the parity sector and together form an *interleave*. When updating a data sector, the corresponding parity sector needs to be updated also. Instead of two read requests, a single longer request involving these two sectors is issued to reduce the response time. The expected length of this single request is evaluated in Section 8.2, where it is also shown that the parity sectors should be placed in the center of the IPC segment to minimize the expected length of this single request.

An IPC scheme with m ($m \leq \ell/2$) interleaves per segment, i.e. ℓ/m sectors per interleave, has the capability of correcting a single error per interleave. Consequently, a segment is in error if there is at least one interleave in which there are at least two unrecoverable sector errors. Note that this scheme can correct a single burst of m consecutive errors occurring in a segment. However, unlike the RS scheme, it in general does not have the capability of correcting any m sector errors in a segment, implying that $P_{\text{seg}}^{\text{IPC}} > P_{\text{seg}}^{\text{RS}}$.

According to the independent model, the probability $P_{\text{interleave}}$ of an interleave

being in error is given by

$$\begin{aligned} P_{\text{interleave}} &= \sum_{j=2}^{\ell/m} \binom{\ell/m}{j} P_s^j (1 - P_s)^{\ell/m-j} = \\ &= \frac{\ell}{m} \frac{(\frac{\ell}{m} - 1)}{2} P_s^2 + O(P_s^3) \approx \frac{\ell(\ell - m)}{2m^2} P_s^2. \end{aligned} \quad (31)$$

Consequently,

$$P_{\text{seg}}^{\text{IPC}} = 1 - (1 - P_{\text{interleave}})^m \approx \frac{\ell(\ell - m)}{2m} P_s^2. \quad (32)$$

In the case of the correlated model, an approximate expression for the probability of a segment being in error is given by the following theorem.

THEOREM 6.8. *It holds that*

$$P_{\text{seg}}^{\text{IPC}} = c_1^{\text{IPC}} P_s + O(P_s^2), \quad (33)$$

where

$$c_1^{\text{IPC}} = 1 + \frac{(\ell - m - 1)G_{m+1} - \sum_{j=1}^m G_j}{B}. \quad (34)$$

PROOF. According to Proposition 6.2, coefficient c_1 is derived based on the probability that a segment contains a single burst of errors and is in error. In the case of the IPC coding scheme, the segment is in error when the burst length exceeds m , which is the same as in the case of the RS scheme. Consequently, the coefficient c_1 is the same as in the case of the RS scheme, i.e. $c_1^{\text{IPC}} = c_1^{\text{RS}}$. \square

COROLLARY 6.9. *For small values of P_s , it holds that*

$$P_{\text{seg}}^{\text{IPC}} \approx \left[1 + \frac{(\ell - m - 1)G_{m+1} - \sum_{j=1}^m G_j}{B} \right] P_s. \quad (35)$$

Remark 6.10. From (26) and (35), it follows that $P_{\text{seg}}^{\text{IPC}} \approx P_{\text{seg}}^{\text{RS}}$ given that $P_{\text{seg}}^{\text{IPC}} - P_{\text{seg}}^{\text{RS}} = O(P_s^2)$. Therefore, when the unrecoverable sector errors are known to occur in bursts whose length can exceed m with a nonnegligible likelihood, using an IPC check code is preferable because it is as efficient as the more complex RS code. This is because the interleaved coding scheme provides additional gain by recovering from consecutive unrecoverable sector errors, which can be as many as the interleaving depth. Note also that if, contrary to our assumption, the maximum burst length does not exceed m , then the term in brackets is equal to zero, implying that $P_{\text{seg}}^{\text{RS}}$ and $P_{\text{seg}}^{\text{IPC}}$ are no longer of order $O(P_s)$. In this case, the two probabilities are of order $O(P_s^2)$ and significantly different.

6.4 Numerical Results

We consider SATA drives with $C_d = 300$ GB and $P_{\text{bit}} = 10^{-14}$. Assuming a sector size of 512 bytes and according to (4), the equivalent unrecoverable sector error probability is $P_s \approx P_{\text{bit}} \times 4096$, which is 4.096×10^{-11} . We also consider a segment comprised of $\ell = 128$ sectors with $m = 8$ interleaves and the following error-burst

Table IV. Approximate P_{seg} .

Coding Scheme	Model for Errors	
	Independent	Correlated
None	5.2×10^{-9}	5.0×10^{-9}
RS	6.2×10^{-81}	2.5×10^{-12}
SPC	1.3×10^{-17}	9.5×10^{-11}
IPC	1.6×10^{-18}	2.5×10^{-12}

Table V. Approximate $P_{\text{uf}}^{(1)}$ for RAID 5 with $N = 8$.

Coding Scheme	Model for Errors	
	Independent	Correlated
None	1.5×10^{-1}	1.5×10^{-1}
RS	2.0×10^{-73}	7.9×10^{-5}
SPC	4.3×10^{-10}	3.1×10^{-3}
IPC	5.1×10^{-11}	7.9×10^{-5}

Table VI. Approximate $P_{\text{uf}}^{(2)}$ for RAID 6 with $N = 16$.

Coding Scheme	Model for Errors	
	Independent	Correlated
None	2.8×10^{-1}	2.7×10^{-1}
RS	3.9×10^{-73}	1.6×10^{-4}
SPC	8.7×10^{-10}	6.1×10^{-3}
IPC	1.0×10^{-10}	1.7×10^{-4}

length distribution:

$$\mathbf{b} = [0.9812 \ 0.016 \ 0.0013 \ 0.0003; 0.0003 \ 0.0002 \ 0.0001 \ 0.0001 \ 0 \ 0.0001 \ 0 \ 0.0001 \ 0.0001 \ 0 \ 0.0001 \ 0.0001] . \quad (36)$$

Then, we have bursts of at most 16 sectors with $\bar{B} = 1.0291$, $G_2 = 0.0188$, and $G_9 = 0.0005$. These values are based on actual data collected from the field for a product that is currently being shipped. The results for P_{seg} are listed in Table IV. The corresponding unrecoverable failure probabilities for a RAID 5 array with $N = 8$ and a RAID 6 array with $N = 16$ are listed in Table V and Table VI, respectively. From the results it follows that in the case of correlated errors, the proposed IPC scheme improves the unrecoverable failure probability by two orders of magnitude compared with the SPC scheme. This is also the improvement we would achieve when using the more complex RS code.

7. RELIABILITY ANALYSIS

In this section, the reliability of a RAID 5 array is analyzed using a direct probabilistic approach. Then, an alternative approach based on a continuous-time Markov chain (CTMC) model is presented and applied to obtain the MTDL for a RAID 6 array. Assuming that the MTDL of a single array is exponentially distributed,

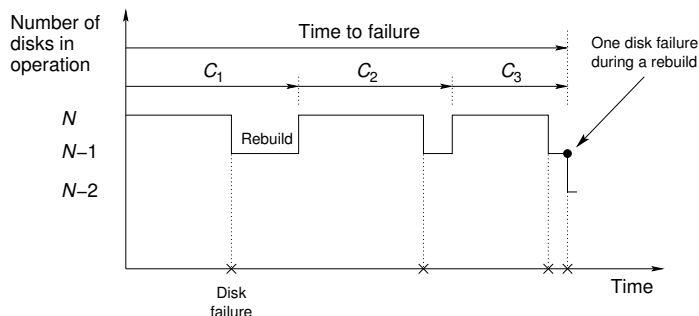


Fig. 5. RAID 5 array operation with normal mode and rebuild cycles.

the MTDDL of a RAID system, $MTDDL_{\text{sys}}$, comprising n_G arrays is subsequently obtained as follows:

$$MTDDL_{\text{sys}} = \frac{MTDDL}{n_G}. \quad (37)$$

7.1 Reliability of RAID 5 Array

The period of safe operation T_G of an array group consists of a number, say M , of cycles $C_1, \dots, C_i, \dots, C_M$, with cycle C_i ($1 \leq i \leq M$) consisting of a normal operation interval T_i followed by a subsequent critical mode interval R_i in which the rebuild process takes place. Thus, $C_i = T_i + R_i$ (see Figure 5). The former interval ends when a disk fails, whereas the latter interval ends when either the rebuild finishes or there is another disk failure during the rebuild phase.

We assume that disk failures are independent and exponentially distributed with parameter λ . Then a RAID 5 array with N disks operating in normal mode experiences the first disk failure after a period that is exponentially distributed with parameter $N\lambda$. Thus, $E(T_i) = 1/N\lambda$. Let F denote the time to the next disk failure while in critical mode. Then F is exponentially distributed with parameter $(N-1)\lambda$, given that now there are $N-1$ disks operating in normal mode. Let us also assume that the rebuild time R in critical mode is exponentially distributed with parameter μ . Then the duration of a critical mode is equal to the minimum of F and R , which in turn is exponentially distributed with parameter $(N-1)\lambda + \mu$, implying that $E(R_i) = \frac{1}{(N-1)\lambda + \mu}$.

Furthermore, the probability P_{fr} that the critical mode ends because of another disk failure is given by

$$P_{\text{fr}} = P(F < R) = \int_0^{\infty} P(F < R | R = x) f_R(x) dx = \frac{(N-1)\lambda}{(N-1)\lambda + \mu}. \quad (38)$$

Note that P_{fr} is also the probability that any cycle is the last one. Consequently, the probability $P(M = k)$ that the period of safe operation consists of k ($k \geq 1$) cycles is equal to $(1 - P_{\text{fr}})^{k-1} P_{\text{fr}}$, as there are $k-1$ successful rebuilds followed by a failed one. Consequently, the random variable M has a geometric distribution with mean $1/P_{\text{fr}}$, i.e. $E(M) = 1/P_{\text{fr}}$. From the above, it follows that the mean time in

each cycle is now given by

$$E(C_i) = E(T_i) + E(R_i) = \frac{1}{N\lambda} + \frac{1}{(N-1)\lambda + \mu}, \quad (39)$$

and that the MTDDL of the RAID 5 array is given by

$$MTDDL = E\left(\sum_{i=1}^M C_i\right) = E(M)E(C_i). \quad (40)$$

Combining (38), (39) and (40), we get

$$MTDDL = \frac{(2N-1)\lambda + \mu}{N(N-1)\lambda^2}. \quad (41)$$

Note that in the case where $\lambda \ll \mu$, (41), together with (37), leads to the expression (2) derived in [Chen et al. 1994; Patterson et al. 1988].

7.2 Unrecoverable Errors and Disk Failures

Let P_{fhr} denote the probability that the critical mode ends because of either another disk failure or an unrecoverable error. Then, the probability $1 - P_{\text{fhr}}$ of the critical mode ending with a successful rebuild is equal to the product of $1 - P_{\text{fr}}$, the probability of not encountering a disk failure during a rebuild, and $1 - P_{\text{uf}}^{(1)}$, the probability of not encountering an unrecoverable error during the rebuild, i.e. $1 - P_{\text{fhr}} = (1 - P_{\text{uf}}^{(1)})(1 - P_{\text{fr}})$. Consequently,

$$P_{\text{fhr}} = P_{\text{uf}}^{(1)} + (1 - P_{\text{uf}}^{(1)})P_{\text{fr}} \quad (42)$$

Analogously to the derivation of (41) and using P_{fhr} instead of P_{fr} , we get

$$MTDDL = \frac{(2N-1)\lambda + \mu}{N\lambda[(N-1)\lambda + \mu P_{\text{uf}}^{(1)}]}. \quad (43)$$

7.3 Continuous-Time Markov Chain (CTMC) Models

Continuous-time Markov models (CTMC) have been extensively used for the reliability analysis of RAID systems [Burkhard and Menon 1993; Malhotra and Trivedi 1993]. Here we establish that the reliability of RAID systems in the presence of unrecoverable errors can also be obtained using CTMC models. Appropriate CTMC models are developed, and, furthermore, we show that these models are also suitable to analyze the reliability of RAID systems that operate in conjunction with an intra-disk redundancy scheme.

First, we demonstrate that the MTDDL for a RAID 5 array derived in Section 7.2 can also be obtained using a CTMC model under the assumptions made in Sections 7.1 and 7.2 regarding the disk failure, unrecoverable error, and rebuild processes. Based on this, we subsequently use the CTMC methodology to obtain the MTDDL for a RAID 6 array. The numbered states of the Markov models represent the number of failed disks. The DF and UF states represent a data loss due to a disk failure and an unrecoverable sector failure, respectively.

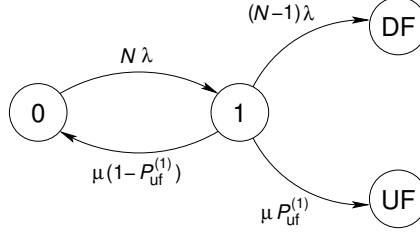


Fig. 6. Reliability model for a RAID 5 array.

7.3.1 *Intra-Disk Redundancy with RAID 5.* In a RAID 5 array, when the first disk fails, the disk array enters the critical mode. This is reflected by the transition from state 0 to state 1 in the Markov chain model shown in Figure 6. The critical mode ends because of either another disk failure (state transition from state 1 to state DF), or a failed rebuild due to an unrecoverable failure (state transition from state 1 to state UF), or a successful rebuild (state transition from state 1 to state 0). As the probability of an unrecoverable failure in the critical mode is $P_{\text{uf}}^{(1)}$, the transition rates from state 1 to states UF and 0 are $\mu P_{\text{uf}}^{(1)}$ and $\mu(1 - P_{\text{uf}}^{(1)})$, respectively.

The infinitesimal generator matrix \mathbf{Q} is given by

$$\begin{bmatrix} -N\lambda & N\lambda & 0 & 0 \\ \mu(1 - P_{\text{uf}}^{(1)}) & -\mu - (N-1)\lambda & (N-1)\lambda & \mu P_{\text{uf}}^{(1)} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

In particular, the submatrix corresponding to the transient states 0 and 1 is

$$\mathbf{Q}_T = \begin{bmatrix} -N\lambda & N\lambda \\ \mu(1 - P_{\text{uf}}^{(1)}) & -\mu - (N-1)\lambda \end{bmatrix}.$$

The vector τ of the average time spent in the transient states before a failure occurs, i.e. before the Markov chain enters either one of the absorbing states DF and UF, is obtained based on the following relation [Trivedi 2002]

$$\tau \mathbf{Q}_T = -\mathbf{P}_T(0),$$

where $\tau = [\tau_0 \ \tau_1]$ and $\mathbf{P}_T(0) = [1 \ 0]$. Solving the above equation for τ yields

$$\tau_0 = \frac{(N-1)\lambda + \mu}{N\lambda[(N-1)\lambda + \mu P_{\text{uf}}^{(1)}]}, \quad \tau_1 = \frac{1}{(N-1)\lambda + \mu P_{\text{uf}}^{(1)}}. \quad (44)$$

Finally, the MTDDL is given by

$$MTTDL = \tau_0 + \tau_1 = \frac{(2N-1)\lambda + \mu}{N\lambda[(N-1)\lambda + \mu P_{\text{uf}}^{(1)}]}, \quad (45)$$

where $P_{\text{uf}}^{(1)}$ is given by (8). Note that this is the same result as in (43). Note also that for $P_{\text{uf}}^{(1)} = 0$ (which holds when $P_s = 0$) and $\lambda \ll \mu$, Eq. (45) can be

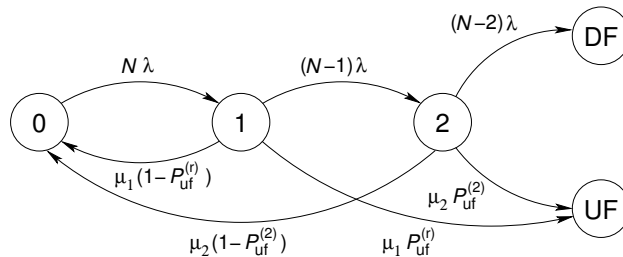


Fig. 7. Reliability model for a RAID 6 array.

approximated as follows:

$$MTTDL \approx \frac{\mu}{N(N-1)\lambda^2}, \quad (46)$$

which is the same result as in (2) (for a single array, i.e. $n_G = 1$).

7.3.2 Intra-Disk Redundancy with RAID 6. A RAID 6 array can tolerate up to two disk failures; thus it is in the critical mode when the disk array has two disk failures. When the first disk fails, the disk array enters into the degraded mode, in which the rebuild of the failing disk takes place while still serves I/O requests. The rebuild of a segment of the failed drive is performed based on up to $N - 1$ corresponding segments residing on the remaining disks. When the rebuild fails, then two or more of these segments are in error. Note, however, that the converse does not hold. It may well be that two segments are in error and the corresponding sectors in error are in such positions that the RAID 6 reconstruction mechanism can correct all of them. Consequently, the probability P_{ref} that a given segment of the failed disk cannot be reconstructed is upper-bounded by the probability that two or more of the corresponding segments residing in the remaining disks are in error. As segments residing in different disks are independent, the upper bound $P_{\text{ref}}^{\text{UB}}$ of the probability P_{ref} is given by

$$P_{\text{ref}}^{\text{UB}} = \sum_{j=2}^{N-1} \binom{N-1}{j} P_{\text{seg}}^j (1 - P_{\text{seg}})^{N-1-j} \approx \binom{N-1}{2} P_{\text{seg}}^2. \quad (47)$$

Furthermore, the reconstruction of each of the n_d segments of the failed disk is independent of the reconstruction of the other segments of this disk. Consequently, the upper bound $P_{\text{uf}}^{(r)}$ of the probability that an unrecoverable failure occurs because the rebuild of the failed disk cannot be completed is given by

$$P_{\text{uf}}^{(r)} = 1 - (1 - P_{\text{ref}}^{\text{UB}})^{n_d}, \quad (48)$$

where n_d is given by (6).

Assuming that the rebuild times in the degraded and the critical mode are exponentially distributed with parameters μ_1 and μ_2 , respectively, we obtain the CTMC model shown in Figure 7. Note that, in contrast to the case of a RAID 5 array, the rate from state 1 to UF is $\mu_1 P_{\text{uf}}^{(r)}$ instead of $\mu_1 P_{\text{uf}}^{(1)}$.

The infinitesimal generator submatrix \mathbf{Q}_T , restricted to the transient states 0, 1

and 2, is given by

$$\begin{bmatrix} -N\lambda & N\lambda & 0 \\ \mu_1(1 - P_{\text{uf}}^{(r)}) & -(N-1)\lambda - \mu_1 & (N-1)\lambda \\ \mu_2(1 - P_{\text{uf}}^{(2)}) & 0 & -(N-2)\lambda - \mu_2 \end{bmatrix}.$$

Solving the equation $\tau \mathbf{Q}_T = -\mathbf{P}_T(0)$ for $\tau = [\tau_0 \ \tau_1 \ \tau_2]$, with $\mathbf{P}_T(0) = [1 \ 0 \ 0]$, we get

$$\tau_0 = \frac{[(N-1)\lambda + \mu_1][(N-2)\lambda + \mu_2]}{N\lambda V}, \quad (49)$$

$$\tau_1 = \frac{(N-2)\lambda + \mu_2}{V}, \quad \tau_2 = \frac{(N-1)\lambda}{V}, \quad (50)$$

where

$$V \triangleq [(N-1)\lambda + \mu_1 P_{\text{uf}}^{(r)}][(N-2)\lambda + \mu_2 P_{\text{uf}}^{(2)}] + \mu_1 \mu_2 P_{\text{uf}}^{(r)}(1 - P_{\text{uf}}^{(2)}), \quad (51)$$

and $P_{\text{uf}}^{(r)}$ and $P_{\text{uf}}^{(2)}$ are given by (48) and (9), respectively.

Then, we have

$$MTTDL = \tau_0 + \tau_1 + \tau_2. \quad (52)$$

Note that for $P_{\text{uf}}^{(r)} = P_{\text{uf}}^{(2)} = 0$ (which holds when $P_s = 0$) and $\lambda \ll \mu_1 = \mu_2 = \mu$, Eq. (52) can be approximated as follows:

$$MTTDL \approx \frac{\mu^2}{N(N-1)(N-2)\lambda^3}, \quad (53)$$

which is the same result as that reported in [Chen et al. 1994].

8. PERFORMANCE EVALUATION

8.1 I/O Performance Analysis

The two key components that make up the response time for an I/O request to a disk are the seek time and the access time [Ruemmler and Wilkes 1994]. The seek time depends on the current and the desired position of the disk head and is typically specified using an average value corresponding to a seek that requires the head to move half of the maximum possible movement. The access time depends on the size of the data unit requested. The response time is determined by the type of workload (e.g., random vs. sequential I/O) and the size of the data unit. The response time of an I/O request, containing e.g. k 4 KB chunks, normalized to the seek time is expressed by the I/O equivalent metric, denoted by IOE , which was introduced in [Hafner et al. 2004]. It is given by

$$IOE = 1 + k/50. \quad (54)$$

For RAID 5 arrays, writing small (e.g., 4 KB) chunks of data located randomly on the disk poses a challenge, the so-called ‘‘small-write’’ problem. This is because each write operation to data also requires the corresponding RAID parity to be updated. A practical way to do this is to read the old data and the old parity from the two corresponding disks, compute the new parity, and then write the new data

and the new parity. Hence, each small-write request results in four I/O requests being issued. A RAID 6 array must update two parity units for each data unit being written. This leads to six I/O requests, namely, reading of the old data and two old parity units, and writing of the new data and the two new parity units. Because of the small size of the data units involved, the predominant component of the response time for each I/O request is the seek time. Based on the above, it follows that the corresponding response times for RAID 5 and RAID 6 expressed through the I/O equivalent metric are given by

$$IOE = \begin{cases} 4(1 + n/400) & \text{for RAID 5} \\ 6(1 + n/400) & \text{for RAID 6,} \end{cases} \quad (55)$$

where n is the I/O request size expressed in sectors.

Using the intra-disk redundancy scheme requires that the intra-disk parity must also be updated whenever a data unit is written. This imposes some constraints on the design of intra-disk redundancy schemes. For a long write, it is natural to directly compute the new intra-disk parity from the new data, and write it along with the data to the disk, resulting in large I/O request lengths and thus longer access time. For a small write, a practical solution is to read the old data and the corresponding old intra-disk parity as part of a single I/O request. Then the new data and the new intra-disk parity are computed and subsequently written back to the disk by a single I/O request. The size of the requested data increases, thereby increasing the access time. However, for small writes and an appropriately designed intra-disk redundancy scheme, the response time is still dominated by the seek time. The issues of the requested data increase and of the placement of the intra-disk parity sectors are addressed in the following subsection.

The scheme proposed in [Hughes and Murray 2004] does not discuss placement of the intra-disk parity sectors. Furthermore, their scheme adds a parity sector for a very large number of data sectors. Therefore, a small-write request must issue separate I/O requests for updating the data and the corresponding intra-disk parity, bringing the total I/O requests to eight. This has an adverse impact on the overall throughput performance.

8.2 Impact of Intra-Disk Redundancy on I/O Performance

Here we analyze the performance of the IPC scheme. We evaluate the average length of a single-sector write when the IPC scheme is used. As mentioned above, when sector A needs to be written, it will be written by a single I/O request also containing the corresponding intra-disk parity sector P_A , as shown in Figure 8 with the parity sectors placed in the p -th row. In fact, the single I/O request will contain all the sectors between A and P_A , as depicted in Figure 8 by the shaded sectors. Note that the total number of sectors n_i depends only on the row position i of sector A , but not on its column position. It is given by

$$n_i = 1 + m \cdot |p - i| \quad \text{for } i \neq p. \quad (56)$$

Therefore, the expected length \bar{n} of a single-sector write request is given by

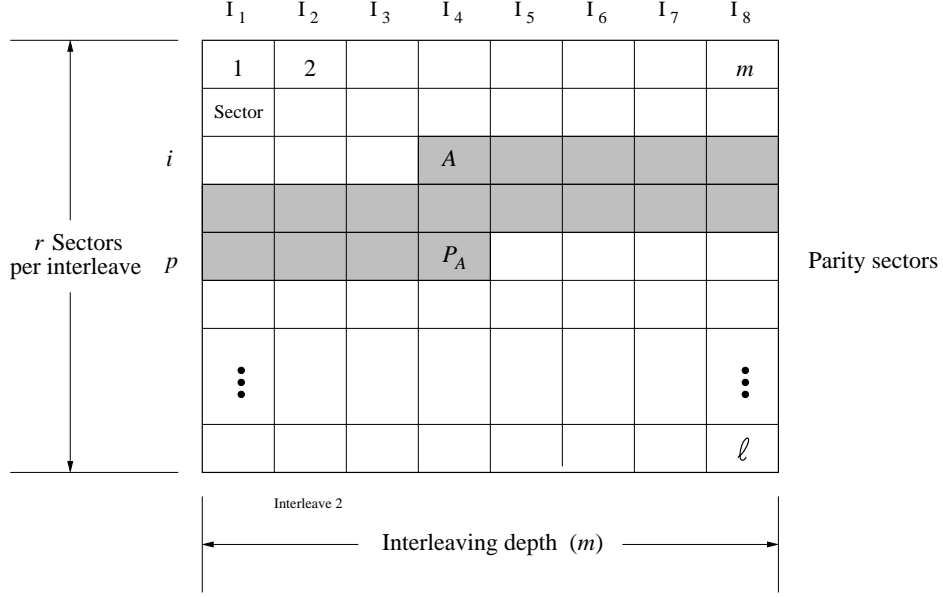


Fig. 8. Length of a single-sector write request using the IPC coding scheme.

$$\bar{n} = \frac{1}{r-1} \sum_{\substack{i=1 \\ i \neq p}}^r n_i = \frac{1}{r-1} \left(\sum_{i=1}^{p-1} n_i + \sum_{i=p+1}^r n_i \right), \quad (57)$$

where

$$r \triangleq \frac{\ell}{m}, \quad r \in \mathbb{N}. \quad (58)$$

Substituting (56) into (57) yields

$$\bar{n} = \frac{1}{r-1} \left[mp^2 - (r+1)mp + \frac{mr(r+1)}{2} + r - 1 \right]. \quad (59)$$

From the above, it follows that \bar{n} is minimized when

$$p = \left\lfloor \frac{r+1}{2} \right\rfloor = \left\lfloor \frac{\ell+m}{2m} \right\rfloor, \quad (60)$$

which implies that the average length of an I/O request is minimized when the the intra-disk parity sectors are placed in the middle of the segment. Substituting (60) into (59) yields

$$\bar{n} = \begin{cases} 1 + \frac{mr^2}{4(r-1)} = 1 + \frac{\ell^2}{4(\ell-m)} & \text{for } r \text{ even} \\ 1 + \frac{m(r+1)}{4} = 1 + \frac{\ell+m}{4} & \text{for } r \text{ odd.} \end{cases} \quad (61)$$

Table VII. I/O Equivalent for Small and Long Writes ($\ell = 128, m = 8$).

Request Length	RAID Scheme	Intra-disk Redundancy Scheme		Relative Difference
		None	IPC	
Small write (1 Sector)	RAID 5	4.01	4.351	8.5 %
	RAID 6	6.015	6.527	8.5 %
Long write (480 Sectors)	RAID 5	8.8	9.12	3.6 %
	RAID 6	13.2	13.68	3.6 %

From (55), it now follows that the corresponding *IOE* metrics for RAID 5 and RAID 6 are given by

$$IOE = \begin{cases} 4(1 + \bar{n}/400) & \text{for RAID 5} \\ 6(1 + \bar{n}/400) & \text{for RAID 6.} \end{cases} \quad (62)$$

Equations (61) and (62) imply that the larger the segment size and the interleaving depth, the higher the *IOE* metric. Let us now consider an IPC scheme with a segment length of 128 sectors, using 8 redundant sectors for every 120 data sectors. This corresponds to $\ell = 128, m = 8, r = 16$, and $p = 8$. The corresponding *IOE* metrics for RAID 5 and RAID 6 are obtained from (62) and listed in Table VII. In the case of no-coding \bar{n} is equal to 1, whereas in the case of IPC coding \bar{n} is derived from (61) and is equal to 35.13. It follows that the response time is dominated by the seek time and that the introduction of the IPC scheme causes the response time for a single-sector I/O request to increase by approx. 9%. In contrast, for a long request, taken to be equal to 480 sectors, the increase is much less, approx. 4%, as shown in Table VII. The corresponding length including the intra-disk parity sectors is 512 sectors. The corresponding *IOE* for no-coding and IPC coding are derived from (62) by setting $\bar{n} = 480$ and $\bar{n} = 512$, respectively. From the results shown in Table VII, we also deduce that a plain RAID 6 system has an I/O performance penalty of 50% compared with a plain RAID 5 system, and approx. 40% compared with a RAID 5 + IPC system.

9. NUMERICAL EXAMPLES

9.1 Analytical Results

Here we assess the reliability of the various schemes considered through illustrative examples. We consider different systems using SATA disk drives and storing a user database of 10 PB. The corresponding parameter values for the SATA disks are summarized in Table VIII. In particular, for a sector size of 512 bytes, we have $P_s = 4.096 \times 10^{-11}$.

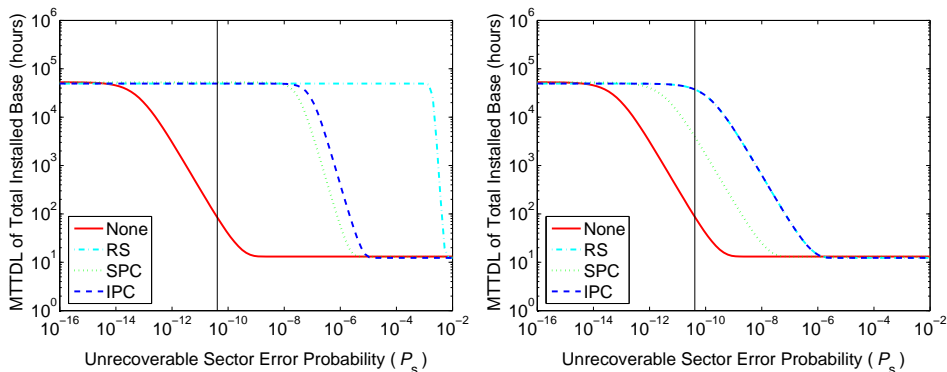
From (10), (11), and (12), it follows that the storage efficiency of the entire system is independent of the RAID configuration if the arrays in a RAID 6 system are twice the size of these in a RAID 5 system. For a RAID 5 system with $N = 8$, when no intra-disk redundancy is used, the required number of arrays to store the user data is equal to 4762 (i.e. $10 \text{ PB}/(7 \times 300 \text{ GB})$), whereas for a RAID 6 system with $N = 16$, it is equal to 2381 (i.e. $10 \text{ PB}/(14 \times 300 \text{ GB})$). The corresponding storage efficiency is equal to $7/8$, i.e. 0.875. For the RS, SPC, and IPC coding schemes, the intra-disk storage efficiency is obtained from (3) by setting $m = 8, 1,$

Table VIII. Parameter Values

Parameter	Value
$1/\lambda$	500,000 h
C_d	300 GB
P_{bit}	10^{-14}
N	8 (for RAID 5), 16 (for RAID 6)
$1/\mu$	17.8 h
$1/\mu_1$	17.8 h
$1/\mu_2$	17.8 h
S	512 bytes = 4096 bits
ℓ	128 sectors
m	8 interleaves per segment

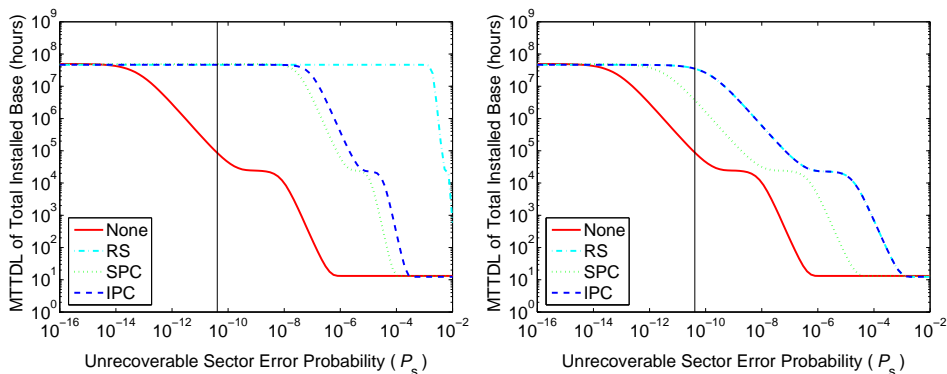
and 8, respectively. For $\ell = 128$, the storage efficiency is equal to 0.94, 0.99, and 0.94, respectively. Furthermore, the required number of arrays for a RAID 5 configuration is obtained as the ratio of 4762 to the intra-disk storage efficiency and is equal to 5080, 4800, and 5080, respectively. Similarly, for a RAID 6 configuration, the required number of arrays is equal to 2540, 2400, and 2540, respectively. The overall storage efficiency is obtained by (12) and is equal to 0.82, 0.87, and 0.82, respectively. Note that the cost of the system is proportional to the required number of arrays and, therefore, inversely proportional to the storage efficiency.

The combined effects of disk and unrecoverable failures can be seen in Figure 9 as a function of the unrecoverable sector error probability. The vertical line in the figures indicates the SATA drive specification for unrecoverable sector errors. Note that for small sector error probabilities, the MTTDL remains unaffected because data is lost owing to a disk rather than an unrecoverable failure. In particular, the MTTDL of a RAID 6 system is three orders of magnitude higher than that of a RAID 5 system. However, as the sector error probability increases, the probability of an unrecoverable failure in the critical mode P_{uf} also increases and therefore the MTTDL decreases. This decrease ends when the sector error probability is such that the corresponding P_{uf} is extremely high, i.e. close to one. In this case the rebuild process in critical mode cannot be successfully completed because of an unrecoverable failure. Consequently, the MTTDL is the mean time until the system (i.e. any of the disk arrays) enters the critical mode. In a RAID 5 system, this occurs when the first disk fails after an expected time of $1/(n_G N \lambda)$. In a RAID 6 system, this occurs when a second disk fails while the system is in the degraded mode. Note that this corresponds to the MTTDL of a RAID 5 system without unrecoverable sector errors. This also explains why the RAID 6 curves become flat at about the height of a RAID 5 system, as can be seen in Figure 10. This range of sector error probabilities is of primary interest because it includes the SATA drive specification. Note that in this range the upper bound $P_{\text{uf}}^{(r)}$ of the probability (as well as the probability itself) of an unrecoverable failure in the degraded mode is negligible, as shown in Figure 11. Consequently, in this range of sector error probabilities, called the first range, the RAID 6 curves are tight lower bounds of the actual MTTDL. We subsequently consider the second range of the remaining sector error probabilities. As the sector error probability further



(a) RAID 5 with independent unrecoverable sector errors.

(b) RAID 5 with correlated unrecoverable sector errors.



(c) RAID 6 with independent unrecoverable sector errors.

(d) RAID 6 with correlated unrecoverable sector errors.

Fig. 9. MTTDL for RAID 5 and RAID 6 systems with unrecoverable sector errors ($\ell = 128$, $m = 8$).

increases, the upper bound $P_{\text{uf}}^{(r)}$ of the probability of an unrecoverable failure in the degraded mode starts becoming significant, as shown in Figure 11, resulting in a further decrease of the MTTDL. This decrease ends when the sector error probability is such that the corresponding $P_{\text{uf}}^{(r)}$ is extremely high, i.e. close to one. In this case the rebuild process in degraded mode cannot be successfully completed because of an unrecoverable failure. Consequently, the MTTDL is the mean time until the system (i.e. any of the disk arrays) enters the degraded mode. In a RAID 6 system, this occurs when the first disk fails after an expected time of $1/(n_G N \lambda)$, which is the same as for a RAID 5 system.

In all cases, the intra-disk redundancy schemes considerably improve the reliability over a wide range of sector error probabilities. In particular, in the case of correlated errors, the IPC coding scheme offers the maximum possible improvement that is also achieved by the RS coding scheme. Furthermore, for large sector error probabilities, the gain from the use of the intra-disk redundancy schemes is smaller

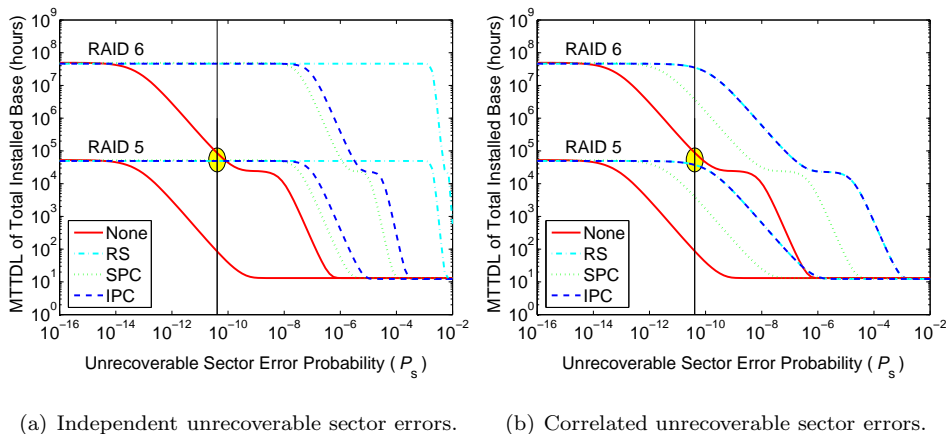


Fig. 10. RAID 5 vs. RAID 6 systems with independent or correlated unrecoverable sector errors ($\ell = 128$, $m = 8$).

for correlated errors than for independent errors. Note that according to Remark 6.10, in the case of correlated errors the MTTDL for the IPC scheme is roughly the same as for the optimum, albeit more complex, RS coding scheme. This is because for both the IPC and RS schemes, and for small sector error probabilities, the probability of an unrecoverable failure is essentially determined by the event of encountering a single burst of more than 8 consecutive errors.

The results shown in Figure 9 along the vertical line reveal that in the practical case of SATA-drive unrecoverable sector errors, the MTTDL is reduced by more than two orders of magnitude. The IPC scheme, however, improves the MTTDL by more than two orders of magnitude, therefore eliminating the negative impact of the unrecoverable sector errors. Note that the IPC scheme can also improve the reliability when disk scrubbing is used. The scrubbing process identifies unrecoverable sector errors at an early stage and attempts to correct them. Data is recovered using the RAID capability, and subsequently written to a good disk location using the bad block relocation mechanism. Thus, the scrubbing effectively reduces the probability of encountering unrecoverable sector errors. The extent of this reduction is a subject of current investigation. If the reduction is of an order of magnitude, then, according to Figure 9, the MTTDL for no coding improves by an order of magnitude. It remains, however, more than an order of magnitude less than the MTTDL offered by the IPC scheme.

Both the plain RAID 6 and the RAID 5 + IPC system improve the reliability over the plain RAID 5 system, with the respective gains shown in Figure 10. Note that in the case of SATA drives the resulting MTTDLs for these two systems are of the same order (depicted by the ellipse) for independent as well as for correlated errors. Therefore, the RAID 5 + IPC system is an attractive alternative to a RAID 6 system, in particular because its I/O performance is better than that of a RAID 6 system, as we shall see in Section 8.2.

We now consider the IPC redundancy scheme employed in conjunction with RAID 5 and RAID 6 systems in the presence of correlated unrecoverable errors

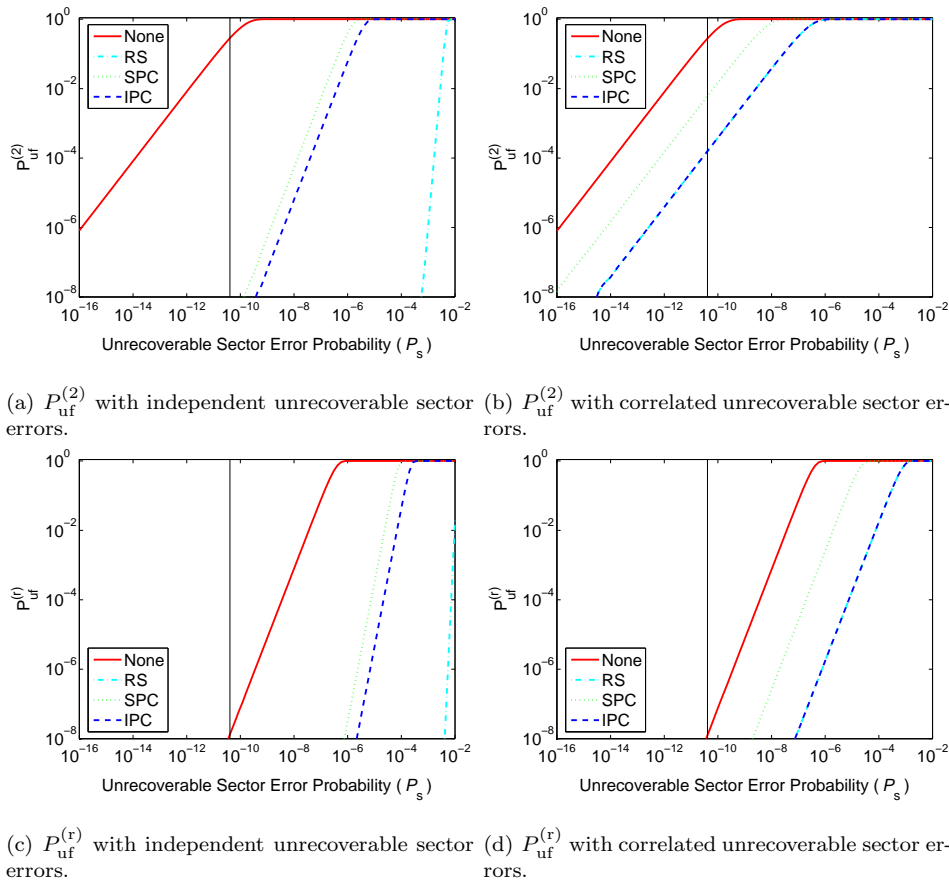


Fig. 11. Probabilities $P_{\text{uf}}^{(2)}$ and $P_{\text{uf}}^{(r)}$ for a RAID 6 system ($\ell = 128, m = 8$).

and investigate the effect of its parameters. First we note that in the range of sector error probabilities of interest, the MTTDL increases as the interleaving depth m increases, as can be seen in Figure 12. This is to be expected because the larger the m the higher the likelihood that a burst of errors can be corrected. In contrast, the MTTDL is practically insensitive to the segment length ℓ , as can be seen in Figure 13, because, regardless of the segment length an unrecoverable failure within a segment is essentially caused by a single burst of errors. A judicious selection of ℓ can be made considering that increasing ℓ results in an increased storage efficiency (i.e. reduced cost), but also in an increased penalty on the I/O performance, according to (12), (61) and (62).

The MTTDL as well as the overall storage efficiency of a RAID 6 system for various values of ℓ and m are shown in Figure 14. We observe that an almost maximal MTTDL is achieved by selecting $m = 8$. This is because when bursts of errors occur, the IPC scheme can correct them in 99.95% of the cases. This percentage corresponds to the probability that the length of a burst does not exceed 8. This reveals that critical to the choice of m and hence to the success of the IPC scheme

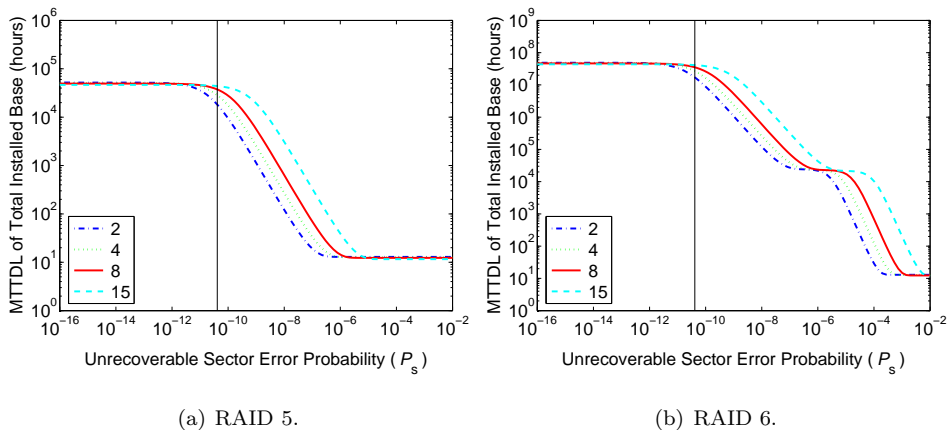


Fig. 12. MTTDL of RAID + IPC systems for $m = 2, 4, 8, 15$ and $\ell = 128$ under correlated unrecoverable sector errors.

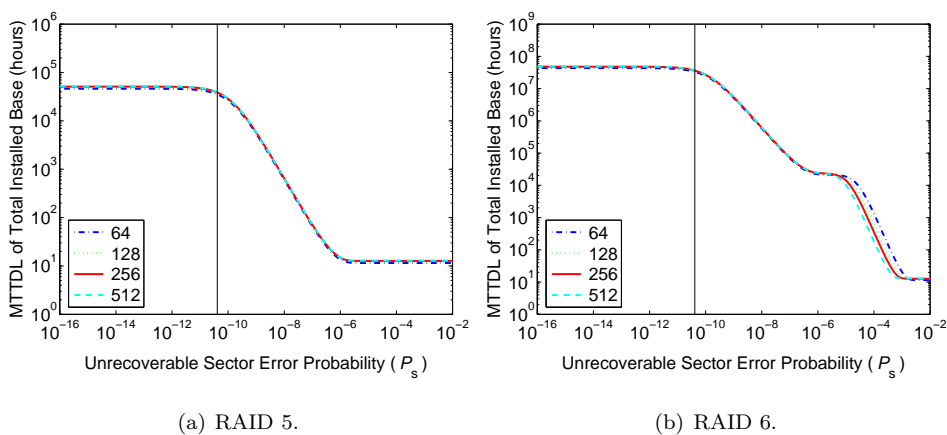


Fig. 13. MTTDL of RAID + IPC systems for $\ell = 64, 128, 256, 512$ and $m = 8$ under correlated unrecoverable sector errors.

is not the actual burst lengths, but rather their probability distribution. According to the results presented in Figure 14, a reasonable compromise between storage efficiency and I/O performance can be achieved by selecting $\ell/m = 16$, which corresponds to $\ell = 128$, $se^{(IDR)} = 94\%$, $se^{(RAID+IDR)} = 82\%$, and a relative I/O performance difference of 8.5% for small writes and 3.6% for long writes, according to the results of Table VII. Note that this is also a good choice of parameter values in the case of a RAID 5 system.

Next we examine the sensitivity of reliability to the error-burst length B by considering a truncated geometric distribution of burst lengths in the range $1, 2, \dots, L$,

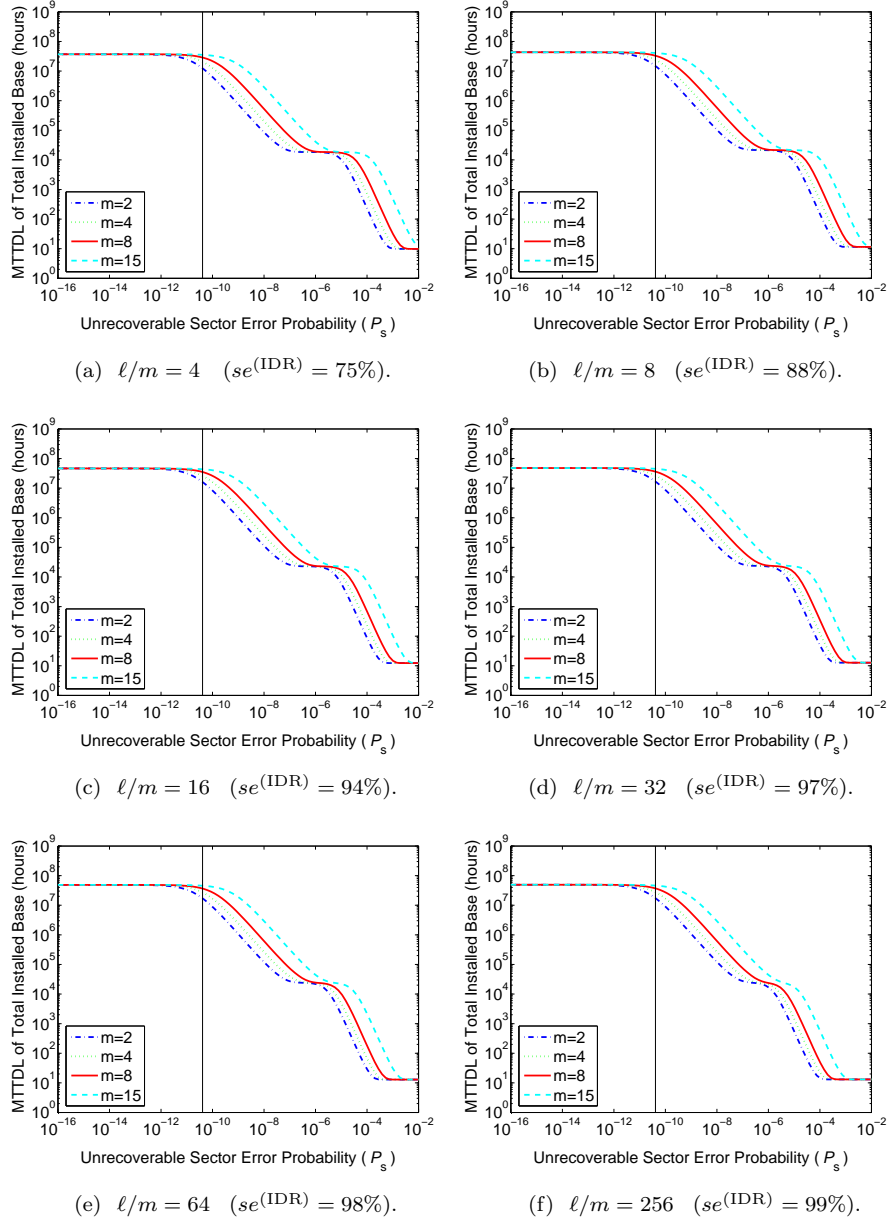


Fig. 14. MTTDL for RAID 6 + IPC systems with correlated unrecoverable sector errors and $m = 2, 4, 8, 15$.

i.e.

$$b_j = \begin{cases} \frac{1-q}{1-q^L} q^{j-1} & \text{for } q \neq 1 \\ \frac{1}{L} & \text{for } q = 1 \end{cases}, \quad \text{with } j = 1, 2, \dots, L, \quad (63)$$

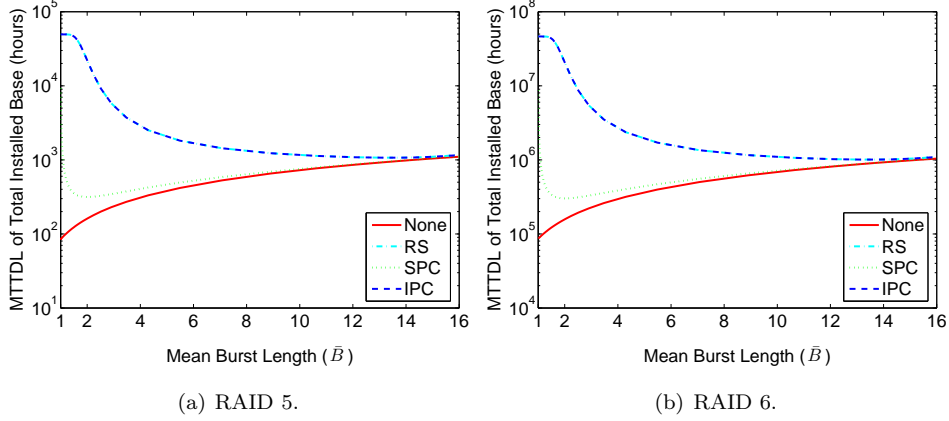


Fig. 15. MTTDL of RAID + IPC systems as a function of \bar{B} for $m = 8$ and $\ell = 128$.

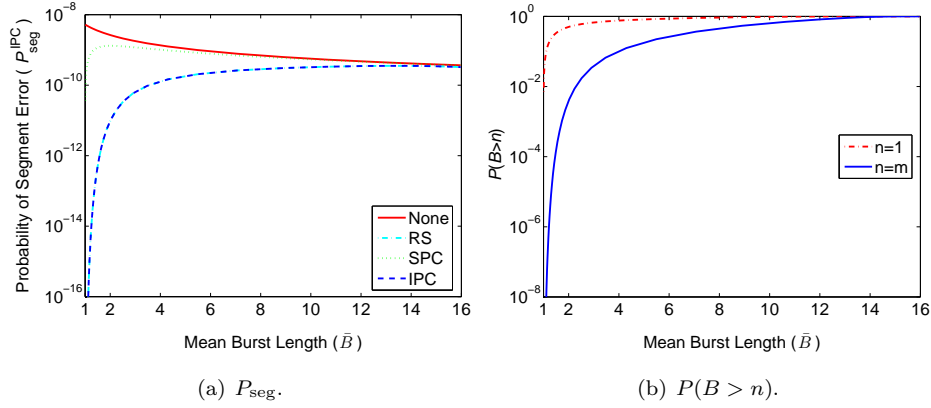


Fig. 16. P_{seg} and $P(B > n)$ as a function of \bar{B} for $m = 8$ and $\ell = 128$.

where q is a parameter taking values in the range $(0, \infty)$. From (63), it follows that the mean burst length \bar{B} is given by

$$\bar{B} = \begin{cases} \frac{1 - (L+1)q^L + Lq^{L+1}}{(1-q)(1-q^L)} & \text{for } q \neq 1 \\ \frac{L}{2} & \text{for } q = 1, \end{cases} \quad (64)$$

which is monotonically increasing in q . Note that for $q = 0$ it holds that $B = \bar{B} = 1$, and that for $q \rightarrow \infty$ it holds that $B = \bar{B} = L$.

Figure 15 shows the MTTDL as a function of the mean burst length for a RAID 5 and a RAID 6 system, and for $P_s = 4.096 \times 10^{-11}$. Clearly, the impact of the mean burst length on the MTTDL is the same for both the RAID 5 and RAID 6 systems. First, we observe that in the case of no coding the MTTDL increases as the mean burst length increases. This is because, owing to (21), $P_{\text{seg}}^{\text{None}}$ decreases in \bar{B} , as can be seen in Figure 16(a). Second, in the case of SPC, the MTTDL drops sharply and

than starts increasing approaching the no-coding case. This is because the SPC scheme improves the reliability significantly only if the probability that the length of an error burst exceeds one is negligible. This holds only when \bar{B} is very small, as can be seen in Figure 16(b). As \bar{B} increases further, the improvement over the no-coding scheme reduces and therefore the MTTDL approaches that of the no-coding scheme. Third, in the case of IPC (and RS) and for small values of \bar{B} , the MTTDL remains unchanged. This is because the probability that the length of an error burst exceeds m is negligible, as can be seen in Figure 16(b), and therefore the IPC scheme can correct almost all of the errors. As \bar{B} increases further, this probability is no longer negligible and, therefore, the effectiveness of the IPC scheme reduces. The improvement over the no-coding scheme reduces and accordingly the MTTDL approaches that of the no-coding scheme. Note that for $B = 1$, there are only single sector errors in a segment, and hence the SPC, IPC and RS schemes are capable of correcting all sector errors. Consequently, the MTTDL is the same as in the case of no coding and $P_s = 0$. The results obtained show that the MTTDL reduction caused by the unrecoverable errors is of more than two orders of magnitude. They also suggest that in the practical case of small mean burst lengths, the IPC scheme copes very efficiently with this problem and therefore results in an improvement of the MTTDL by more than two orders of magnitude.

9.2 Simulation Results

In this section we focus on using event-driven simulation techniques to characterize various redundancy schemes, specifically to study the performance impact of the intra-disk redundancy scheme when incorporated into RAID systems. Two performance metrics are commonly used to benchmark a storage system: the response time and the saturation throughput.

Most modern RAID controllers have a large battery-backed cache that boosts the overall system performance by reducing the I/O requests to the disks, and performing aggressive read-ahead and write-behind. The response time of an array as experienced by the end user can be dramatically shortened by increasing the size of the array cache and selecting the replacement strategy based on the characteristics of workloads. As our main interest in the simulation is the performance difference of RAID schemes rather than caching mechanism or characteristics of workloads, we start measuring the response time of requests after caching, i.e. from the instant when they are sent to the disks. Therefore, the saturation throughput measures the maximum throughput between the front-end (cache) and the back-end (disk array). The higher the saturation throughput, the better the performance of the underlying RAID mechanism.

We have developed a lightweight event-driven simulator that also includes a HDD model, specifically a 3.5-inch SCSI IBM Ultrastar 146Z10 having a capacity of 146.8 GB and a rotational speed of 10K RPM. Various standard RAID simulators are publicly available in the community, such as, for example, HP Labs' Pantheon for disk arrays [HP Labs 2006]. However, these simulators focus mainly on standard RAID functions and are not flexible enough to easily accommodate a new level of redundancy such as we wish. With the advent of the C++ standard library and the concept of generic programming, particularly the standard template library (STL), developing a lightweight event-driven simulator from scratch often turns

out to be an easier task than understanding and tailoring an existing large software package. Another alternative would have been to use the CMU’s DiskSim for disks [The DiskSim Simulation Environment (Version 3.0) 2007], but we found that DiskSim only supports some obsolete disk models. Therefore, we have built an HDD module targeted for the IBM drive 146Z10, following the approach described in [Ruemmler and Wilkes 1994] and consulting the source code of DiskSim. The disk-drive model captures major features such as zoned cylinder allocation, mechanical positioning parameters such as seek time, settling time, cylinder and head skew, as well as rotational latency, data transfer latency, and buffering effects such as read ahead. The simulated response time of the HDD exhibits a good match with its nominal specification. We assume a first-come first-served (FCFS) scheduling policy for serving the I/O requests at each disk. Actually, we have tested several other disk-scheduling policies such as SSTF, LOOK, and C-LOOK, and have found that the scheduling policy practically does not affect the performance of the intra-disk redundancy scheme.

We compare the RAID 5 and RAID 6 schemes with the corresponding schemes enhanced by the addition of the intra-disk redundancy scheme. We also consider a RAID N+3 scheme, which is a natural extension of the RAID 5 and RAID 6 schemes that uses three redundant disks to protect against as many as three simultaneous disk failures. In our entire evaluation, each array consists of 8 disks. For the intra-disk redundancy scheme, we employ an IPC scheme with a segment size of 128 sectors, comprising 8 redundant sectors and 120 data sectors.

First we focus on the small-write scenario and use synthetic workloads generating aligned 4 KB small I/O requests with uniformly distributed logical block addresses (LBAs). The ratio of read to write is set to be 1:2, i.e. there are 33.33% reads and 66.67% writes, because a front-end cache reduces the number of read requests sent to the disks. The request inter-arrival times are assumed to be exponentially distributed. Figure 17 shows the average response times for a range of arrival rates. Of primary interest is the mean arrival rate at a given mean response time. It is evident that RAID 6 and RAID N+3 suffer severely from the small-write problem compared with RAID 5, suggesting that they are too costly to cope with unrecoverable failures when these are the predominant source of data loss. In contrast, the RAID 5 and RAID 6 schemes enhanced by the IPC-based intra-disk redundancy scheme exhibit a more graceful degradation. The saturation throughput for RAID 5 is 305 I/O requests per array per second, whereas for the IPC scheme on top of RAID 5 it is 295 I/O requests per array per second. This represents a minor, 3% degradation in saturation throughput due to the IPC scheme. Similarly, a minor degradation in saturation throughput is observed when the IPC scheme is used on top of RAID 6.

In Figure 18 we investigate the impact of having request sizes exponentially distributed with a mean of 256 KB. These requests approximate a mix of random and sequential requests. We set the read-to-write ratio to 2:1. We observe that the relative performance of the five RAID schemes mentioned does not change, although the corresponding differences are reduced. This is to be expected because for read requests and for sequential requests the impact of different redundancy schemes is not as significant as in the case of small updates. The saturation throughput

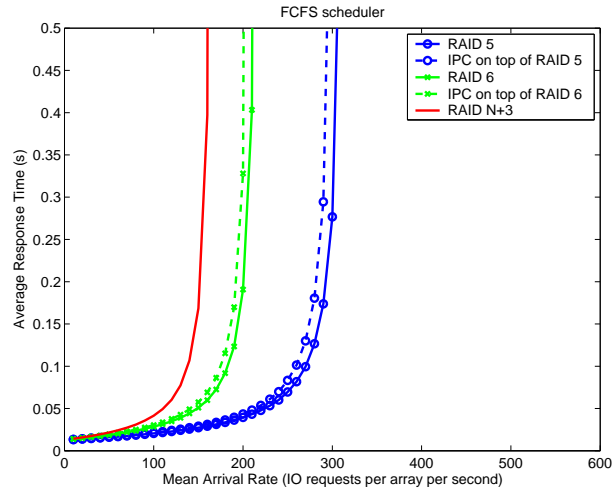


Fig. 17. Response time of various RAID systems (synthetic workload, small writes).

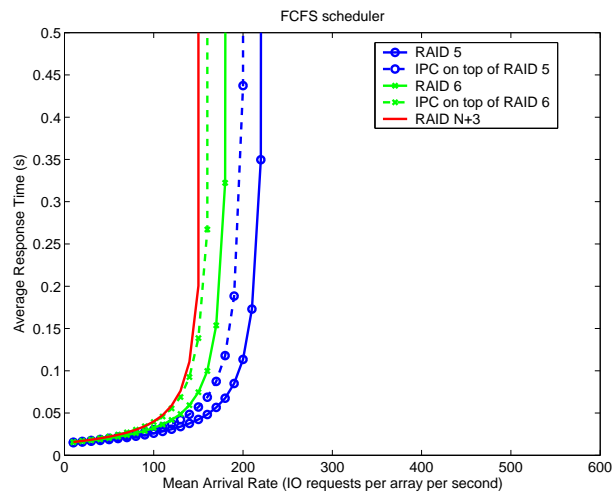


Fig. 18. Response time of various RAID systems (synthetic workload, mix of random and sequential requests).

for RAID 5 is 218 I/O requests per array per second, whereas for the IPC scheme on top of RAID 5 it is 200 I/O requests per array per second. This represents a 9% degradation in saturation throughput due to the IPC scheme. It may seem conceptually counterintuitive that the IPC overhead is smaller for the small-write case than for the large-write case. This is due to the fact that the small-write case is 4K aligned, whereas the large-write case is not.

To gain an understanding of how these redundancy schemes perform under actual user workloads, we use two traces from the Storage Performance Council (SPC) benchmark SPC-1 [SPC a; b] that have the largest data records, namely, the Financial 1 (154 MB) and Websearch 2 (139 MB). The traces vary widely in their

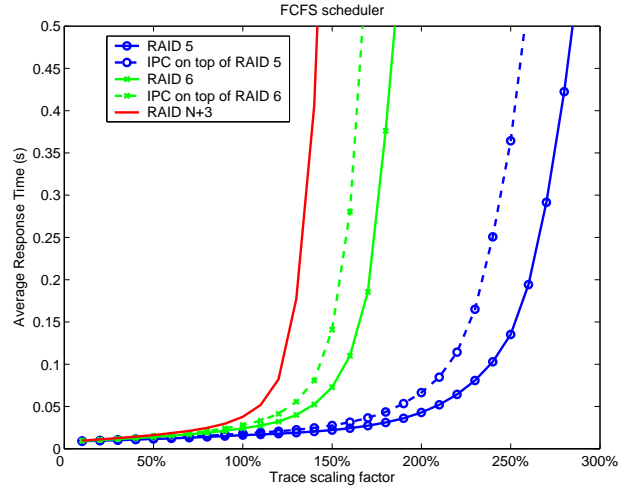


Fig. 19. Response time of various RAID systems (SPC Financial 1 trace).

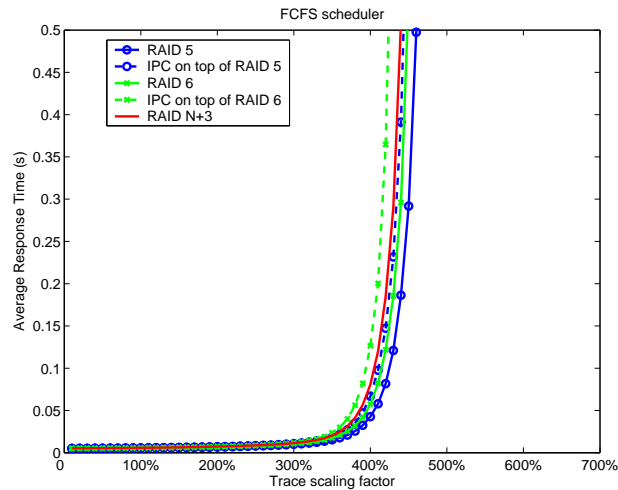


Fig. 20. Response time of various RAID systems (SPC Websearch 2 trace).

read/write ratios, access sizes, arrival rates, degree of sequentiality, and burstiness. The performance graphs use a range of arrival-rate scaling factors for the traces. The workloads with a unity (100%) scaling factor correspond to the original request stream. Figure 19 shows the average response times for a range of trace-scaling factors on the Financial 1 trace. As approx. 76.8% requests are small writes in Financial 1 trace, we observe that the IPC on top of RAID 5 scheme performs slightly worse than the RAID 5 scheme but significantly better than the RAID 6 scheme. Similarly, the IPC on top of RAID 6 scheme performs worse than the RAID 6 scheme but better than the RAID N+3 scheme.

Figure 20 shows the average response times for a range of trace scaling-factors on Websearch 2 trace. This trace is characterized by a nearly 100% reads, with request

sizes ranging from 8 to 32 KB. In the case of the intra-disk redundancy scheme it follows that there is a slight performance drawback due to alignment issues.

10. CONCLUSIONS

Owing to increasing disk capacities and the adoption of low-cost disks in modern data storage systems, unrecoverable errors are becoming a significant cause of user data loss. To cope with this issue, a new intra-disk redundancy scheme was introduced and its design described. An intra-disk redundancy architecture that is specifically based on a simple interleaved parity-check (IPC) coding scheme was proposed. A new model capturing the effect of correlated unrecoverable sector errors was developed to analyze this scheme. Moreover, redundancy schemes based on traditional Reed–Solomon (RS) codes and single-parity-check codes were analyzed in the context of the generic intradisk redundancy architecture. Closed-form expressions were derived for the mean time to data loss of RAID 5 and RAID 6 systems in the presence of unrecoverable errors and disk failures. The I/O and throughput performance of the RAID 5 and RAID 6 systems enhanced by the new intra-disk redundancy scheme was evaluated by means of analysis and simulations. Our results demonstrate that the proposed IPC-based intra-disk redundancy scheme considerably improves reliability over a wide range of sector error probabilities. In particular, in the case of correlated errors, the IPC coding scheme offers the maximum possible improvement that is also achieved by the RS coding scheme. Furthermore, the associated penalty on the I/O performance is minimal. Therefore a RAID 5 system enhanced by an intra-disk redundancy scheme that uses IPC is an attractive alternative to a RAID 6 system as its reliability is similar to and its I/O performance better than that of a RAID 6 system. Alternative designs of the intra-disk redundancy concept introduced in this paper and a potential adoption of other erasure-coding schemes are the subject of further investigation.

APPENDIX

A. NUMBER OF BURSTS OF ERRORS IN A SEGMENT

PROOF OF PROPOSITION 6.1.

Let us consider an instance of k bursts in a segment and let us denote by \vec{L} the vector (L_1, \dots, L_k) of the corresponding burst lengths and by \vec{S} the vector (S_1, \dots, S_k) of their corresponding starting sector positions with $1 \leq S_1 < \dots < S_k \leq \ell$. The length of the error-free interval I_j following the j -th burst is then given by $S_{j+1} - S_j - L_j$, for $j = 1, 2, \dots, k-1$. Also, the length of the error-free interval I_0 preceding the first burst is at least $S_1 - 1$, and the length of the error-free interval I_k following the k -th burst is at least $\ell + 1 - S_k - L_k$.

Let us now consider the following realization in terms of burst lengths $\vec{l} = (l_1, \dots, l_k)$ and starting sector positions $\vec{s} = (s_1, \dots, s_k)$. Let us denote by \mathcal{R}_k the set of all possible realizations $\{(\vec{l}, \vec{s})\}$, and by \mathcal{E}_k its subset containing those realizations that lead to a segment error. Next we proceed to calculating the probability $P(\vec{L} = \vec{l}, \vec{S} = \vec{s})$. Depending on the value of s_1 , two cases are considered:

Case 1) $s_1 = 1$. As the first sector of the segment has an error, the corresponding burst may have started in the preceding segment. Therefore, the length R_1 of the remaining consecutive errors is distributed according to the residual burst length

\hat{B} , i.e. $P(R_1 = j) = \hat{b}_j$, where $\hat{b}_j \triangleq P(\hat{B} = j) = G_j/\bar{B}$ for $j = 1, 2, \dots$ [Kleinrock 1976]. Note that the length L_1 of consecutive errors within the segment is equal to $\min(R_1, \ell)$, and therefore its pdf is given by $P(L_1 = j) = P(R_1 = j) = \hat{b}_j$ for $j = 1, 2, \dots, \ell - 1$, and $P(L_1 = \ell) = P(R_1 \geq \ell) = \sum_{j=\ell}^{\infty} \hat{b}_j$. Depending on whether I_k exists, two cases are considered:

Case 1.a) $\exists I_k$. This is equivalent to the condition $s_k + l_k \leq \ell$. As in this case the length of the interval I_k is at least $\ell + 1 - s_k - l_k$, it holds that

$$P(\vec{L} = \vec{l}, \vec{S} = \vec{s}) = P(\text{first sector in error}, L_1 = l_1, I_1 = s_2 - s_1 - l_1, L_2 = l_2, \dots, L_k = l_k, I_k \geq \ell + 1 - s_k - l_k) = P_s P(L_1 = l_1) P(I_1 = s_2 - s_1 - l_1) P(L_2 = l_2) \cdots P(L_k = l_k) P(I_k \geq \ell + 1 - s_k - l_k) = P_s \frac{G_{l_1}}{B} \alpha(1 - \alpha)^{s_2 - s_1 - l_1 - 1} b_{l_2} \cdots b_{l_k} (1 - \alpha)^{\ell - s_k - l_k} = P_s \frac{G_{l_1}}{B} b_{l_2} \cdots b_{l_k} (1 - \alpha)^{\ell - k - (l_1 + \dots + l_k)} \alpha^{k-1} = \frac{G_{l_1} b_{l_2} \cdots b_{l_k}}{B^k} P_s^k + O(P_s^{k+1}).$$

Case 1.b) $\nexists I_k$. This is equivalent to the condition $s_k + l_k = \ell + 1$. Depending on the value of k , two cases are considered:

Case 1.b.i) $k = 1$. In this case it holds that $l_1 = \ell$. Thus,

$$P(L_1 = \ell, S_1 = 1) = P(\text{first sector in error}, R_1 \geq \ell) = P_s P(R_1 \geq \ell) = \frac{\sum_{j=\ell}^{\infty} G_j}{B} P_s.$$

Case 1.b.ii) $k \geq 2$. As the last sector of the segment has an error, the corresponding burst may extend into the next segment. Therefore, the pdf of the length L_k of consecutive errors within the segment is distributed according to the complementary cumulative density function of the burst length B , i.e. $P(L_k = n) = \sum_{j=n}^{\infty} b_j = G_n$ for $n = 1, 2, \dots$. In this case it holds that $s_k + l_k = \ell + 1$. Thus,

$$P(\vec{L} = \vec{l}, \vec{S} = \vec{s}) = P(\text{first sector in error}, L_1 = l_1, I_1 = s_2 - s_1 - l_1, L_2 = l_2, \dots, L_k = l_k) = P_s P(L_1 = l_1) P(I_1 = s_2 - s_1 - l_1) P(L_2 = l_2) \cdots P(I_{k-1} = s_k - s_{k-1} - l_{k-1}) P(L_k = l_k) = P_s \frac{G_{l_1}}{B} \alpha(1 - \alpha)^{s_2 - s_1 - l_1 - 1} b_{l_2} \cdots \alpha(1 - \alpha)^{s_k - s_{k-1} - l_{k-1} - 1} G_{l_k} = P_s \frac{G_{l_1}}{B} b_{l_2} \cdots b_{l_{k-1}} G_{l_k} (1 - \alpha)^{\ell - (k-1) - (l_1 + \dots + l_k)} \alpha^{k-1} = \frac{G_{l_1} b_{l_2} \cdots b_{l_{k-1}} G_{l_k}}{B^k} P_s^k + O(P_s^{k+1}).$$

Case 2) $s_1 \geq 2$. Let P_{bs} be the probability that a burst of errors starts at a given sector position. This is equal to the product of the probability of the sector being in error and of the probability of an erroneous sector being the first of its corresponding burst, i.e. $P_{\text{bs}} = P_s/\bar{B}$. Depending on whether I_k exists, two cases are considered:

Case 2.a) $\exists I_k$. This is equivalent to the condition $s_k + l_k \leq \ell$. Similarly to Case 1.a, it holds that

$$P(\vec{L} = \vec{l}, \vec{S} = \vec{s}) = P(I_0 \geq s_1 - 1, \text{burst of errors starts at } s_1, L_1 = l_1, I_1 = s_2 - s_1 - l_1, \dots, L_k = l_k, I_k \geq \ell + 1 - s_k - l_k) = P(I_0 \geq s_1 - 1) P_{\text{bs}} P(L_1 = l_1) P(I_1 = s_2 - s_1 - l_1) \cdots P(L_k = l_k) P(I_k \geq \ell + 1 - s_k - l_k) = (1 - \alpha)^{s_1 - 2} \frac{P_s}{B} b_{l_1} \alpha(1 - \alpha)^{s_2 - s_1 - l_1 - 1} \cdots b_{l_k} (1 - \alpha)^{\ell - s_k - l_k} = \frac{P_s}{B} b_{l_1} \cdots b_{l_k} (1 - \alpha)^{\ell - k - (l_1 + \dots + l_k) - 1} \alpha^{k-1} = \frac{b_{l_1} \cdots b_{l_k}}{B^k} P_s^k + O(P_s^{k+1}).$$

Case 2.b) $\nexists I_k$. This is equivalent to the condition $s_k + l_k = \ell + 1$. Similarly to Case 1.b.ii, and for all values of k , it holds that

$$P(\vec{L} = \vec{l}, \vec{S} = \vec{s}) = P(I_0 \geq s_1 - 1, \text{burst of errors starts at } s_1, L_1 = l_1, I_1 = s_2 - s_1 - l_1, \dots, L_k = l_k) = P(I_0 \geq s_1 - 1) P_{\text{bs}} P(L_1 = l_1) P(I_1 = s_2 - s_1 - l_1) \cdots P(I_{k-1} = s_k - s_{k-1} - l_{k-1}) P(L_k = l_k) = (1 - \alpha)^{s_1 - 2} \frac{P_s}{B} b_{l_1} \alpha(1 -$$

$$\alpha^{s_2-s_1-l_1-1} \cdots \alpha(1-\alpha)^{s_k-s_{k-1}-l_{k-1}-1} G_{l_k} = \frac{P_s}{B} b_{l_1} \cdots b_{l_{k-1}} G_{l_k} \quad (1 - \alpha)^{\ell-k-(l_1+\cdots+l_k)} \alpha^{k-1} = \frac{b_{l_1} \cdots b_{l_{k-1}} G_{l_k}}{B^k} P_s^k + O(P_s^{k+1}).$$

From the above, it follows that $P(\vec{L} = \vec{l}, \vec{S} = \vec{s})$ is of order $O(P_s^k)$ because for every (\vec{l}, \vec{s}) it holds that $P(\vec{L} = \vec{l}, \vec{S} = \vec{s}) = \frac{A(\vec{l}, \vec{s})}{B^k} P_s^k + O(P_s^{k+1})$, with $A(\vec{l}, \vec{s})$ being a function of \vec{l} , \vec{s} and $\{b_j\}$. Consequently, $P_{\text{seg}}^{(k)} = \sum_{(\vec{l}, \vec{s}) \in \mathcal{E}_k} P(\vec{L} = \vec{l}, \vec{S} = \vec{s}) = \frac{\sum_{(\vec{l}, \vec{s}) \in \mathcal{E}_k} A(\vec{l}, \vec{s})}{B^k} P_s^k + O(P_s^{k+1})$. \square

B. REED-SOLOMON (RS) CODING SCHEME

PROOF OF THEOREM 6.3.

According to Proposition 6.2, coefficient c_1 is derived based on the probability that a segment contains a single burst of errors and is in error. In the case of an RS coding scheme, the segment is in error when the burst length exceeds m . Consequently, for $k = 1$ and using the terminology of Appendix A, the segment is in error for all realizations (l, s) such that $l \geq m + 1$. Thus,

$$P_{\text{seg}} = \sum_{\substack{l \geq m+1 \\ 1 \leq i \leq \ell}} P(L = l, S = i) = \sum_{l=m+1}^{\ell-1} P(L = l, S = 1) + P(L = \ell, S = 1) + \\ + \sum_{i=2}^{\ell-m-1} \sum_{l=m+1}^{\ell-i} P(L = l, S = i) + \sum_{i=2}^{\ell-m} P(L = \ell + 1 - i, S = i),$$

with the four summation terms corresponding to Cases 1.a, 1.b.i, 2.a and 2.b, respectively. Using the following relations $\bar{B} = \sum_{j=1}^{\infty} G_j$ and $b_j = G_j - G_{j+1}$, $j \in \mathbb{N}$, we get

$$P_{\text{seg}} = \sum_{l=m+1}^{\ell-1} \frac{G_l}{B} P_s + \frac{\sum_{j=\ell}^{\infty} G_j}{B} P_s + \sum_{i=2}^{\ell-m-1} \sum_{l=m+1}^{\ell-i} \frac{b_l}{B} P_s + \sum_{i=2}^{\ell-m} \frac{G_{\ell+1-i}}{B} P_s + O(P_s^2) = \\ = \left(\sum_{l=m+1}^{\infty} G_l + \sum_{i=2}^{\ell-m-1} \sum_{l=m+1}^{\ell-i} b_l + \sum_{i=2}^{\ell-m} G_{\ell+1-i} \right) \frac{P_s}{B} + O(P_s^2) = \\ = \left(\sum_{l=1}^{\infty} G_l - \sum_{l=1}^m G_l + \sum_{l=m+1}^{\ell-2} \sum_{i=2}^{\ell-l} b_l + \sum_{i=m+1}^{\ell-1} G_i \right) \frac{P_s}{B} + O(P_s^2) = \\ = \left[1 + \frac{(\ell-m-1)G_{m+1} - \sum_{j=1}^m G_j}{B} \right] P_s + O(P_s^2).$$

\square

REFERENCES

- BLAUM, M., BRADY, J., BRUCK, J., AND MENNON, J. 1995. EVENODD: an efficient scheme for tolerating double disk failures in RAID architectures. *IEEE Trans. Comput.* 44, 2 (Feb.), 192–202.
- BURKHARD, W. A. AND MENON, J. 1993. Disk array storage system reliability. In *Proceeding of the 23rd International Symposium on Fault-Tolerant Computing* (Toulouse, France). 432–441.
- CHEN, P. M., LEE, E., GIBSON, G., KATZ, R., AND PATTERSON, D. 1994. RAID: High-performance, reliable secondary storage. *ACM Computing Surveys* 26, 2 (June), 145–185.

- CHEN, S. AND TOWSLEY, D. 1996. A performance evaluation of RAID architectures. *IEEE Trans. Comp.* 45, 10 (Oct.), 1116–1130.
- CORBETT, P., ENGLISH, R., GOEL, A., GRACANAC, T., KLEIMAN, S., LEONG, J., AND SANKAR, S. 2004. Row-diagonal parity for double disk failure correction. In *Proceedings of the 3rd USENIX Conference on File and Storage Technologies (FAST)* (San Francisco, CA). 1–14.
- HAFNER, J. L., DEENADHAYALAN, V., KANUNGO, T., AND RAO, K. 2004. Performance metrics for erasure codes in storage systems. Research Report RJ 10321, IBM Almaden Research Center.
- HITACHI GLOBAL STORAGE TECHNOLOGIES. 2007. Hitachi disk drive product datasheets. <http://www.hitachigst.com/>.
- HP LABS. 2006. Private Software. http://tesla.hpl.hp.com/private_software/.
- HUGHES, G. F. AND MURRAY, J. F. 2004. Reliability and security of RAID storage systems and D2D archives using SATA disk drives. *ACM Trans. Storage* 1, 1 (Dec.), 95–107.
- KEETON, K., SANTOS, C., BEYER, D., CHASE, J., AND WILKES, J. 2004. Designing for disasters. In *Proceedings of the 3rd USENIX Conference on File and Storage Technologies (FAST)* (San Francisco, CA). 59–72.
- KLEINROCK, L. 1976. *Queueing Systems, Volume 2: Computer Applications*. Wiley, New York.
- MALHOTRA, M. AND TRIVEDI, K. S. 1993. Reliability analysis of redundant arrays of inexpensive disks. *J. Parallel Distrib. Comp.* 17, 146–151.
- MALHOTRA, M. AND TRIVEDI, K. S. 1995. Data integrity analysis of disk array systems with analytic modeling of coverage. *Performance Evaluation* 22, 111–133.
- PATTERSON, D. A., GIBSON, G., AND KATZ, R. H. 1988. A case for redundant arrays of inexpensive disks (RAID). In *Proceedings of the ACM SIGMOD International Conference on Management of Data* (Chicago, IL). 109–116.
- RUEMLER, C. AND WILKES, J. 1994. An introduction to disk drive modeling. *IEEE Computer*, 17–28.
- SCHULZE, M., GIBSON, G., KATZ, R., AND PATTERSON, D. 1989. How reliable is a RAID? In *Proceedings of the 34th IEEE COMPCON* (San Francisco, CA). 118–123.
- SPC. Storage performance council, storage OLTP application I/O traces. <http://prisms.cs.umass.edu/repository/>.
- SPC. Storage performance council, storage search engine I/O traces. <http://prisms.cs.umass.edu/repository/>.
- The DiskSim Simulation Environment (Version 3.0) 2007. <http://www.pdl.cmu.edu/DiskSim/>.
- TRIVEDI, K. S. 2002. *Probabilistic and Statistics with Reliability, Queueing and Computer Science Applications, 2nd Ed.* Wiley, New York.
- VARKI, E., MERCHANT, A., XU, J., AND QIU, X. 2004. Issues and challenges in the performance analysis of real disk arrays. *IEEE Trans. Par. Distr. Syst.* 15, 6 (June), 559–574.
- WU, X., LI, J., AND KAMEDA, H. 1997. Reliability analysis of disk array organizations by considering uncorrectable bit errors. In *Proceedings of the 16th IEEE Symposium on Reliable Distributed Systems* (Durham, NC). 2–9.
- XIN, Q., MILLER, E. L., SCHWARZ, T., LONG, D. D. E., BRANDT, S. A., AND LITWIN, W. 2003. Reliability mechanisms for very large storage systems. In *Proceedings of the 20th IEEE/11th NASA Goddard Conference on Mass Storage Systems and Technologies (MSST)* (San Diego, CA). 146–156.