RZ 3652 (# 99662) 03/16/2006
Computer Science 13 pages

Research Report

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

Ajay Dholakia,[1] Evangelos Eleftheriou,[2] Xiao-Yu Hu,[2] Ilias Iliadis,[2] Jai Menon,[3] and KK Rao[4]

[1]IBM Systems and Technology Group, Research Triangle Park, NC
[2]IBM Research, Zurich Research Laboratory, Rüschlikon, Switzerland
[3]IBM Systems and Technology Group, San Jose, CA
[4]IBM Almaden Research Center, San Jose, CA

E-mail: adholak@us.ibm.com, {ele, xhu, ili}@zurich.ibm.com, menonjm@almaden.ibm.com, kkrao@us.ibm.com

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

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

Ajay Dholakia[1] Evangelos Eleftheriou[2] Xiao–Yu Hu[2] Ilias Iliadis[2] Jai Menon[3] KK Rao[4]

[1] IBM Systems and Technology Group, Research Triangle Park, NC
[2] IBM Research, Zurich Research Laboratory, Rüschlikon, Switzerland
[3] IBM Systems and Technology Group, San Jose, CA
[4] IBM Almaden Research Center, San Jose, CA

adholak@us.ibm.com, {ele, xhu, ili}@zurich.ibm.com, menonjm@almaden.ibm.com, kkrao@us.ibm.com

ABSTRACT

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 a new XOR-based intra-disk redundancy scheme, called interleaved parity check (IPC), to enhance the reliability of RAID systems that incurs only negligible I/O performance degradation. The proposed 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, while the proposed scheme aims to protect against media-related unrecoverable errors. A comparison between the proposed scheme and traditional redundancy schemes based on Reed-Solomon (RS) codes and single-parity-check (SPC) 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. Furthermore, we 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 for a comprehensive characterization of the reliability of RAID systems that incorporate the proposed intra-disk redundancy scheme. Our results show that in the practical case of correlated errors, the proposed scheme provides the same reliability as the optimum albeit more complex RS coding scheme. Finally, the throughput performance of incorporating the intra-disk redundancy on various RAID systems is evaluated by means of event-driven simulations.

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 HDD 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) [3, 13]. 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 stripe units per set of data stripe units [1, 5]. 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, which is 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 stripe unit is partitioned into segments, and within each segment, a portion of the storage, usually several sectors, is used for storing data, called data sectors, 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 XOR-based intra-disk redundancy scheme is introduced for erasure correction in the presence of unrecoverable sector errors. The novelty of this scheme is the combination of XOR-based parity, interleaving, and parity placement, all put together in a practical implementation. Furthermore, a new model capturing the effect of correlated unrecoverable sector errors is developed and subsequently used to analyze the proposed 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 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 which incorporate the proposed intra-disk redundancy scheme. Finally, the throughput performance of these RAID systems is evaluated using event-driven simulations under a variety of workloads.

As our results demonstrate, the easy-to-implement interleaved parity-check coding scheme proposed 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 new scheme is that it can be applied to various RAID systems, including RAID 5 and RAID 6.

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, including a detailed analysis of the erasure correction capability in the presence of independent as well as correlated unrecoverable sector errors. Section 5 is devoted to assessing the reliability of RAID 5 and RAID 6 storage systems that incorporate the proposed coding scheme. In Section 6, the throughput performance is evaluated using event-driven simulations to address dynamic I/O. Section 7 concludes the paper.

2. RELATED WORK

Data storage systems are being designed to meet increasingly more stringent data integrity requirements [10]. 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 the low-cost alternative to the 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 [8]. A simple scheme based on using intra-disk redundancy is described in [7] 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 [13], the reliability of RAID systems was analyzed by several groups. A basic reliability analysis of RAID systems was presented in [2, 11, 15]. Unrecoverable errors were considered in [12], 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 [20], 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 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 [21]. 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 [4], the performance of different RAID systems was studied and various scheduling policies were presented. More recently, an integrated performance model was developed in [19] 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:

- 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 SCSI and 17.8 h for SATA drives.
- Unrecoverable bit error probability (P_{bit}): 1×10^{-15} for SCSI and 1×10^{-14} for SATA drives.
- Drive capacity (C_d in bytes): SCSI drives with 73, 146, and 300 GB, and SATA drives with 300 and 500 GB.

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 on 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

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

Fig. 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 mean time to data loss (MTTDL) metric. In the presence of disk failures only, the MTTDL of a RAID 5 array is well known [3, 13] and is given by

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

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 Fig. 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

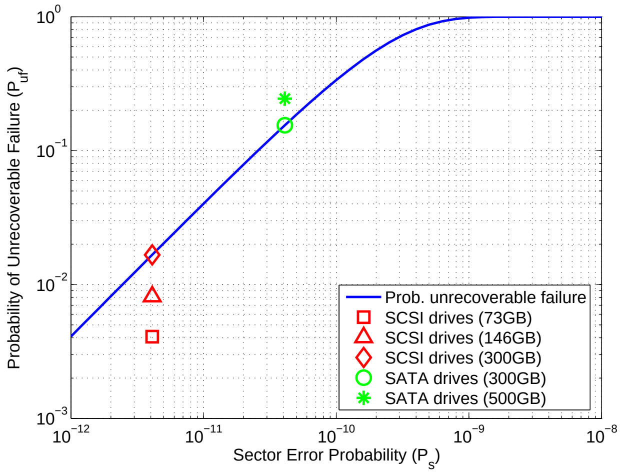

Figure 1: P_{uf} as a function of P_s.

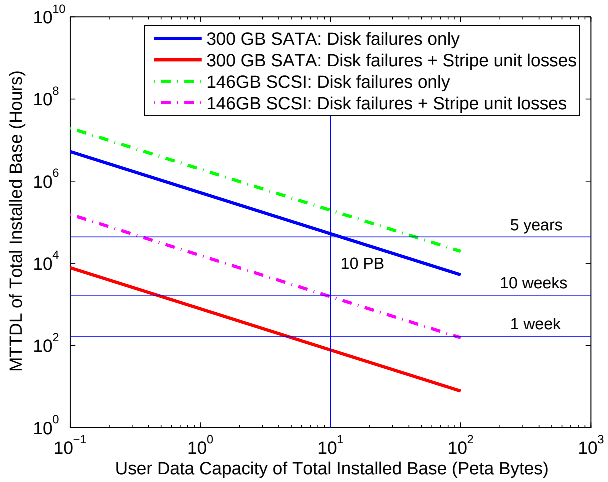

Figure 2: MTTDL of RAID 5 arrays as a function of total installed user capacity.

5.2, a 10 PB installation using 146 GB SCSI drives experiences a MTTDL of around ten weeks, as shown in Fig. 2. More interestingly, the MTTDL of a 10 PB installation using 300 GB SATA drives drops to less than one week. These examples reveal 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 sectors in a stripe unit are grouped together into a segment. Redundant sectors derived from the data sectors are also included in the same segment. A number of different schemes can be used to obtain the redundant parity sectors, as will be described later in this section. The entire segment, comprising ℓ data and parity sectors, is stored contiguously on the same disk, as shown in Fig. 3, where $\ell = n + m$.

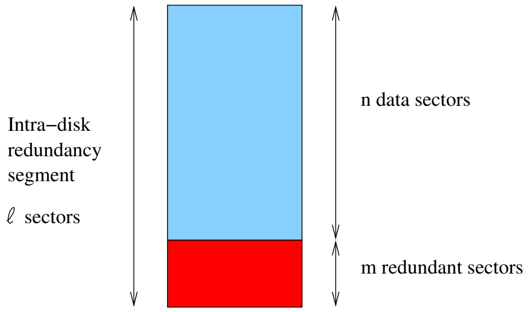

Figure 3: Basic intra-disk redundancy scheme.

The size of a segment should be chosen such that a sufficient degree of storage efficiency, performance and reliability are ensured. In addition, the number 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 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.

4.1 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 $1 - P_s$. 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 iid, i.e. independently and identically distributed random variables. In particular, as in the independent model, the error-free intervals are assumed to be geometrically distributed with a parameter α. Therefore, the pdf $\{a_j\}$ of the length j of a typical error-free interval is given by $a_j = P(I = j) = (1 - \alpha)\alpha^{j-1}$ for $j = 1, 2, \ldots$, such that $\bar{I} = 1/(1 - \alpha)$. Let also $\{b_j\}$ denote the pdf of the length j of a typical burst of consecutive errors, i.e. $Pr(B = j) = b_j$. 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

$$P_s = \bar{B}/(\bar{B} + \bar{I}) \; . \tag{3}$$

From the above it follows that

$$\begin{aligned} \alpha &= 1 - \frac{P_s}{\bar{B}(1 - P_s)} = 1 - \frac{P_s}{\bar{B}} - \frac{P_s^2}{\bar{B}} - \frac{P_s^3}{\bar{B}} - \cdots \\ &= 1 - \frac{P_s}{\bar{B}} - O(P_s^2) \approx 1 - \frac{P_s}{\bar{B}} \; . \end{aligned} \tag{4}$$

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 P_s, i.e. $b_j = (1 - P_s)P_s^{j-1}$ for $j = 1, 2, \ldots$, and $\bar{B} = 1/(1 - P_s)$. Let $\{G_n\}$ denote the complementary cumulative density function (ccdf) of the burst size 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 $n = 1, 2, \ldots$. 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

$$P_{seg} = 1 - (1 - P_s)^{\ell} = \ell P_s + O(P_s^2) \approx \ell P_s \; . \tag{5}$$

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 α. By making use of (4) we get

$$\begin{aligned} P_{seg} &= 1 - (1 - P_s)\alpha^{\ell-1} = 1 - (1 - P_s)\left(1 - \frac{P_s}{\bar{B}}\right)^{\ell-1} \\ &\approx \left(1 + \frac{\ell - 1}{\bar{B}}\right) P_s \; . \end{aligned} \tag{6}$$

We now proceed with the evaluation of P_{seg} for various coding schemes. First we establish the following propositions which hold independently of the coding scheme used.

Proposition 1. The probability $P_{burst,k}$ that a segment contains k ($k \le \ell/2$) bursts is of order $O(P_s^k)$.
Proof: See Appendix A. ■

Proposition 2. It holds that $P_{seg} \approx P$(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 1, we get $P_{seg} = \sum_{k=1}^{\ell/2} P(\text{segment in error}|k \text{ bursts}) P_{burst,k} = \sum_{k}^{\ell/2} P(\text{segment in error }|k \text{ bursts})\, O(P_s^k) \approx P(\text{segment in error }|\text{ single burst})\, O(P_s)$, with the approximation holding iff the probability $P(\text{segment in error }|\text{ single burst})$ is several orders of magnitude larger than P_s. This holds when the probability that the length of a burst exceeds a value m (which depends on the coding scheme used) is substantial, i.e. when $G_{m+1} \gg P_s$. Consequently, to obtain the approximate expression for P_{seg}, it suffices to consider only the case of a single burst within the segment. ■

The two models are now used for the performance evaluation of various coding schemes. In the critical mode, an unrecoverable failure occurs when at least one out of the n_s

segments that need to be read is in error. Consequently, the probability of an unrecoverable failure, $P_{\rm uf}$, is given by

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

For a RAID 5 and a RAID 6 system in the critical mode, the corresponding probabilities of an unrecoverable failure $P_{\rm uf}^{(1)}$ and $P_{\rm uf}^{(2)}$ are obtained by setting $n_s = (N-1)C_d/512\ell$ and $n_s = (N-2)C_d/512\ell$, as there are $N-1$ and $N-2$ operational disks, respectively, i.e.

$$P_{\rm uf}^{(1)} = 1 - (1 - P_{\rm seg})^{\frac{(N-1)C_d}{512\ell}} \ , \qquad (8)$$

and

$$P_{\rm uf}^{(2)} = 1 - (1 - P_{\rm seg})^{\frac{(N-2)C_d}{512\ell}} \ . \qquad (9)$$

We proceed with the evaluation of the probability $P_{\rm seg}$ corresponding to the various coding schemes.

4.2 Reed-Solomon (RS) Coding

Reed-Solomon (RS) coding is the standard choice for erasure correction when the implementation complexity is not a constraint. This is because the 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

$$P_{\rm seg}^{\rm RS} = \sum_{j \ge m+1}^{\ell} \binom{\ell}{j} P_{\rm s}^j (1-P_{\rm s})^{\ell-j} \approx \binom{\ell}{m+1} P_{\rm s}^{m+1} \ . \qquad (10)$$

In the case of the correlated model, and according to Proposition 2, the probability of a segment being in error is roughly equal to the probability of it containing a burst of more than m unrecoverable sector errors. In Appendix B it is shown that

$$P_{\rm seg}^{\rm RS} \approx \left[1 + \frac{(\ell - m - 1)G_{m+1} - \sum_{j=1}^{m} G_j}{\bar{B}}\right] P_{\rm s} \ . \qquad (11)$$

Note that the term in brackets can also be written as $((\ell - m)G_{m+1} + \sum_{j=m+2}^{\infty} G_j)/\bar{B}$, which, owing to the assumptions that $G_{m+1} \gg P_{\rm s}$ and $\bar{B} \ll 1/P_{\rm s}$, is several orders of magnitude larger than $P_{\rm s}$.

4.3 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

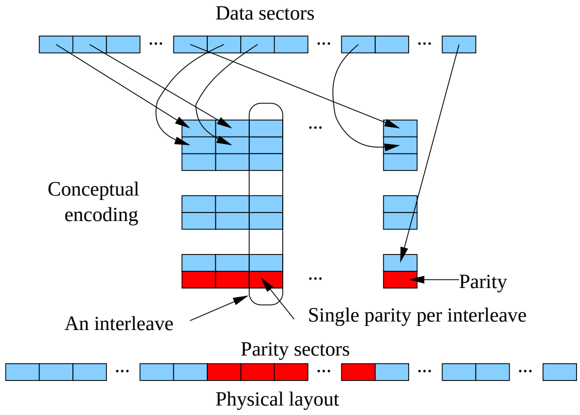

Figure 4: Intra-disk redundancy scheme using the interleaved parity-check coding scheme.

at least two unrecoverable sector errors. The independent model yields

$$P_{\rm seg}^{\rm SPC} = \sum_{j=2}^{\ell} \binom{\ell}{j} P_{\rm s}^j (1-P_{\rm s})^{\ell-j} \approx \frac{\ell(\ell-1)}{2} P_{\rm s}^2 \ . \qquad (12)$$

In the case of the correlated model, and according to Proposition 2, the probability of a segment being in error is roughly equal to the probability of containing a burst of at least two unrecoverable sector errors. The corresponding expression is derived from (11) by setting $m = 1$,

$$P_{\rm seg}^{\rm SPC} \approx \left[1 + \frac{(\ell - 2)G_2 - 1}{\bar{B}}\right] P_{\rm s} \ . \qquad (13)$$

4.4 Interleaved Parity-Check (IPC) Coding

A new coding scheme, called interleaved parity check (IPC), that has a simplicity akin to that of the SPC scheme but considerably better performance is introduced next. In this scheme, n contiguous data sectors are conceptually arranged in a matrix as shown in Fig. 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 as well. Instead of two read requests, a single longer request involving these two sectors is issued to reduce the response time. Consequently, the parity sectors are placed in the center of the IPC segment to minimize the expected length of this single request.

An IPC scheme with m ($m \le \ell/2$) 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 $P_{\rm seg}^{\rm IPC} > P_{\rm seg}^{\rm RS}$.

According to the independent model, the probability $P_{\rm interleave}$ of an interleave being in error is given by

$$P_{\rm interleave} = \sum_{j \ge 2}^{\ell/m} \binom{\ell/m}{j} P_{\rm s}^j (1-P_{\rm s})^{\ell/m-j} \approx \frac{\ell(\ell-m)}{2m^2} P_{\rm s}^2 \ .$$

Table 1: 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 2: Approximate $P_{\text{uf}}^{(1)}$ for RAID 5.

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}

Consequently,

$$P_{\text{seg}}^{\text{IPC}} = 1 - (1 - P_{\text{interleave}})^m \approx \frac{\ell(\ell - m)}{2m}\, P_{\text{s}}^2 \; . \qquad (14)$$

In the case of the correlated model, and according to Proposition 2, the probability of a segment being in error is roughly equal to the probability of it containing a burst of more than m unrecoverable sector errors, which was derived in (11), i.e.

$$P_{\text{seg}}^{\text{IPC}} \approx \left[1 + \frac{(\ell - m - 1)G_{m+1} - \sum_{j=1}^{m} G_j}{\bar{B}} \right] P_{\text{s}} \; . \qquad (15)$$

From (11) and (15) it follows that $P_{\text{seg}}^{\text{IPC}} \approx P_{\text{seg}}^{\text{RS}}$ given that $P_{\text{seg}}^{\text{IPC}} - P_{\text{seg}}^{\text{RS}} = O(P_{\text{s}}^2)$. 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 $O(P_{\text{s}})$. In this case, the two probabilities are of order $O(P_{\text{s}}^2)$ and significantly different.

4.5 Numerical Results

We consider SATA drives with the following set of parameters: $P_{\text{s}} = 4096 \times 10^{-14}$, $\mathbf{b} = [0.9812\ 0.016\ 0.0013\ 0.0003\ 0.0003\ 0.0002\ 0.0001\ 0.0001\ 0\ 0.0001\ 0\ 0.0001\ 0.0001\ 0\ 0.0001\ 0.0001]$, $\ell = 128$ and $m = 8$. Then, we have bursts of at most 16 sectors with $\bar{B} = 1.0291$, $G_2 = 0.0188$, $G_9 = 0.0005$, such that $\min(G_2, G_9, \bar{B}^{-1}) = 0.0005 \gg 4096 \times 10^{-14}$. These values are based on actual data collected from the field for a product that is being shipping. The results for P_{seg} are listed in Table 1. The corresponding unrecoverable failure probabilities for a RAID 5 system with $N = 8$ are listed in Table 2. 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 get by using the more complex RS code.

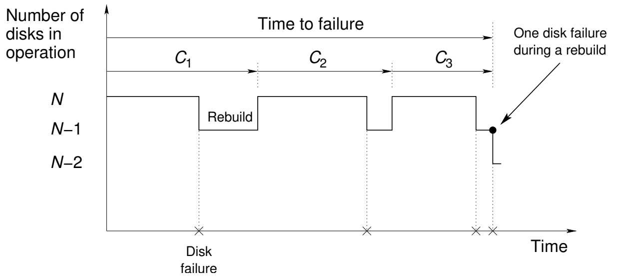

Figure 5: RAID 5 array operation with normal mode and rebuild cycles.

5. 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. This approach is subsequently applied to obtain the MTTDL for a RAID 6 array. Assuming that the MTTDL of a single array is exponentially distributed, the MTTDL of a RAID system, $MTTDL_{\text{sys}}$, comprising n_G arrays is subsequently obtained as follows:

$$MTTDL_{\text{sys}} = \frac{MTTDL}{n_G} \; . \qquad (16)$$

5.1 Reliability of RAID 5 Array

The period of safe operation T_G of an array group consists of a number, say M, of cycles $C_1, \ldots, C_i, \ldots, C_M$, with cycle C_i $(1 \le i \le M)$ 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 Fig. 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

$$\begin{aligned} P_{\text{fr}} &= P(F < R) = \int_0^\infty P(F < R | R = x) f_R(x) dx \\ &= \frac{(N-1)\lambda}{(N-1)\lambda + \mu}. \end{aligned} \qquad (17)$$

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 $(k \ge 1)$ 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. $E(M) = 1/P_{\rm fr}$. From the above it follows that the mean time in each cycle is now given by

$$E(C_i) = E(T_i) + E(R_i) = \frac{1}{N\lambda} + \frac{1}{(N-1)\lambda + \mu} , \quad (18)$$

and that the mean time to data loss of the RAID 5 array is given by

$$MTTDL = E(\sum_{i=1}^{M} C_i) = E(M)E(C_i). \quad (19)$$

Combining (17), (18) and (19), we get

$$MTTDL = \frac{(2N-1)\lambda + \mu}{N(N-1)\lambda^2}. \quad (20)$$

Note that in the case where $\lambda \gg \mu$, (20) leads to the expression (2) derived in [3, 13].

5.2 Unrecoverable Errors and Disk Failures

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

$$P_{\rm fhr} = P^{(1)}_{\rm uf} + (1 - P^{(1)}_{\rm uf})P_{\rm fr} \quad (21)$$

Analogously to the derivation of (20) and using $P_{\rm fhr}$ instead of $P_{\rm fr}$, we get

$$MTTDL = \frac{(2N-1)\lambda + \mu}{N\lambda[(N-1)\lambda + \mu P^{(1)}_{\rm uf}]}. \quad (22)$$

5.3 CTMC Models

Continuous-time Markov models (CTMC) have been extensively used for the reliability analysis of RAID systems [2, 11]. Here we establish that the reliability of RAID systems in the presence of unrecoverable errors can also be obtained using appropriate CTMC models. Furthermore, the CTMC models introduced 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 MTTDL for a RAID 5 array derived in Section 5.2 can also be obtained using a CTMC model under the assumptions made in Sections 5.1 and 5.2 regarding the disk failure, unrecoverable error, and rebuild processes. Based on this, we subsequently use the CTMC methodology to obtain the MTTDL 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.

5.3.1 *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 Fig. 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 $P^{(1)}_{\rm uf}$, the transition rates from state 1 to states UF and 0 are $\mu_1 P^{(1)}_{\rm uf}$ and $\mu_1(1 - P^{(1)}_{\rm uf})$, respectively.

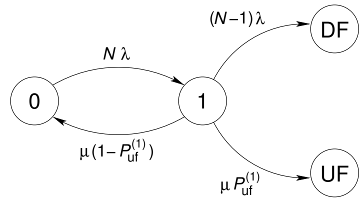

Figure 6: Reliability model for a RAID 5 array.

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

$$\begin{bmatrix} -N\lambda & N\lambda & 0 & 0 \\ \mu(1 - P^{(1)}_{\rm uf}) & -\mu - (N-1)\lambda & (N-1)\lambda & \mu P^{(1)}_{\rm uf} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

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

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

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 [18]

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

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

$$\tau_0 = \frac{(N-1)\lambda + \mu}{N\lambda[(N-1)\lambda + \mu P^{(1)}_{\rm uf}]}, \quad \tau_1 = \frac{N\lambda}{N\lambda[(N-1)\lambda + \mu P^{(1)}_{\rm uf}]}.$$

Finally, the mean time to data loss is given by

$$MTTDL = \tau_0 + \tau_1 = \frac{(2N-1)\lambda + \mu}{N\lambda[(N-1)\lambda + \mu P^{(1)}_{\rm uf}]} , \quad (23)$$

which is the same result as in (22).

5.3.2 *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 serving 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 $P_{\rm recf}$ 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

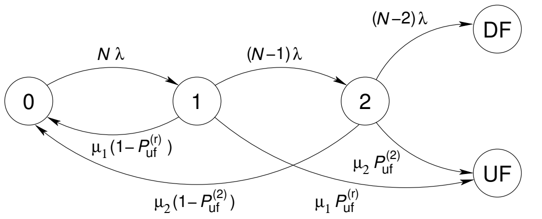

Figure 7: Reliability model for a RAID 6 array.

error. As segments residing in different disks are independent, the upper bound $P_{\mathrm{recf}}^{\mathrm{UB}}$ of the probability P_{recf} is given by

$$P_{\mathrm{recf}}^{\mathrm{UB}} = \sum_{j=2}^{N-1} \binom{N-1}{j} P_{\mathrm{seg}}^{j} (1-P_{\mathrm{seg}})^{N-1-j} \approx \binom{N-1}{2} P_{\mathrm{seg}}^{2} . \quad (24)$$

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 $P_{\mathrm{uf}}^{(r)}$ of the probability that an unrecoverable failure occurs because the rebuild of the failed disk cannot be completed is given by

$$P_{\mathrm{uf}}^{(r)} = 1 - (1 - P_{\mathrm{recf}}^{\mathrm{UB}})^{n_d} , \qquad \text{where} \quad n_d = \frac{C_d}{512\ell} . \quad (25)$$

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 Fig. 7. Note that, in contrast to the case of a RAID 5 array, the rate from state 1 to UF is $\mu_1 P_{\mathrm{uf}}^{(r)}$ instead of $\mu_1 P_{\mathrm{uf}}^{(1)}$.

The infinitesimal generator submatrix $\mathbf{Q}_T$, restricted to the transient states 0, 1 and 2, is given by

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

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

$$\tau_0 = \frac{[(N-1)\lambda + \mu_1]\,[(N-2)\lambda + \mu_2]}{N\lambda V} , \quad (26)$$

$$\tau_1 = \frac{(N-2)\lambda + \mu_2}{V} , \qquad \tau_2 = \frac{(N-1)\lambda}{V} , \quad (27)$$

where

$$V \triangleq [(N-1)\lambda + \mu_1 P_{\mathrm{uf}}^{(r)}][(N-2)\lambda + \mu_2 P_{\mathrm{uf}}^{(2)}] + \mu_1 \mu_2 P_{\mathrm{uf}}^{(r)} (1 - P_{\mathrm{uf}}^{(2)}) , \quad (28)$$

and $P_{\mathrm{uf}}^{(r)}$ and $P_{\mathrm{uf}}^{(2)}$ are given by (25) and (9), respectively.

Then, we have

$$MTTDL = \tau_0 + \tau_1 + \tau_2 . \quad (29)$$

5.4 Numerical Examples

Here we assess the reliability of various schemes considered above through illustrative examples. The combined effects of disk and unrecoverable failures can be seen in Figs. 8, 9, 10 and 11 as a function of the unrecoverable sector error probability. We assume SATA drives with $N = 8$, $n_G = 125000$, $C_d = 300$ GB and $\lambda^{-1} = 5 \times 10^5$ h. For

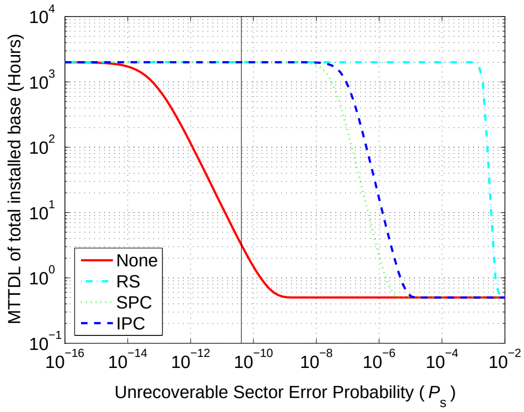

Figure 8: MTTDL for a RAID 5 system with independent unrecoverable sector errors.

a RAID 5 system, $\mu^{-1} = 17.8$ h, and for a RAID 6 system $\mu_1^{-1} = \mu_2^{-1} = 17.8$ h. The vertical line in the figures shows the SATA drive specification for unrecoverable sector errors. 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, the gain from the use of the intra-disk redundancy schemes is smaller in the case of correlated errors compared with independent errors.

6. PERFORMANCE EVALUATION

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

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 [14]. 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 requested data unit. The response time is determined by the type of workload (e.g., random vs. sequential I/O) and the size of the data unit.

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. 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. 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.

Using an intra-disk redundancy scheme, such as IPC, requires that the intra-disk parity must also be updated when-

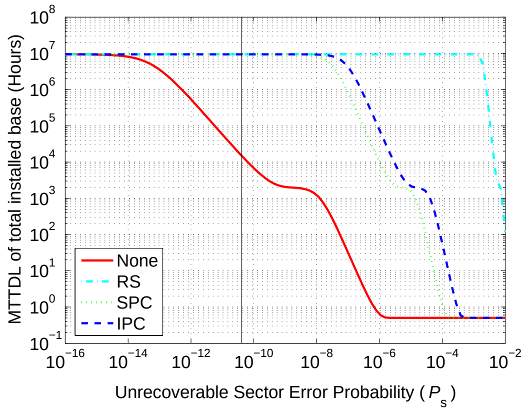

Figure 9: MTTDL for a RAID 6 system with independent unrecoverable sector errors.

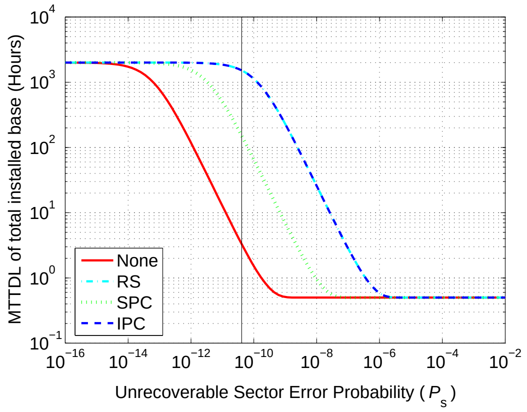

Figure 10: MTTDL for a RAID 5 system with correlated unrecoverable sector errors.

ever a data unit is written. This imposes some constraints on the design of intra-disk redundancy schemes. A practical solution is to read the old data and the corresponding old intra-disk parity as part of 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 impact of the increased access time can be taken into account to obtain an accurate value for the response time. For an IPC scheme with 8 redundant sectors for each 120 data sectors, the response time for each of the four I/O requests increases by approx. 10%.

The scheme proposed in [7] does not discuss placement of the intra-disk parity sectors. Furthermore, the suggested 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.

6.2 Simulation Results

In this section we focus on using event-driven simulation techniques to characterize various redundancy schemes, particularly to study the performance impact of the intra-disk redundancy scheme when incorporated into RAID systems. Three performance metrics are commonly used to benchmark a storage system, namely, response time, saturation throughput, and queue length. Response time is the time spent by an I/O request at the disk array, from its inception to the completion of service. The size of the memory of the RAID controller should be dimensioned to accommodate the queue lengths that build up. The mean queue length can be obtained by Little's law as the product of the mean response time and the mean arrival rate or throughput.

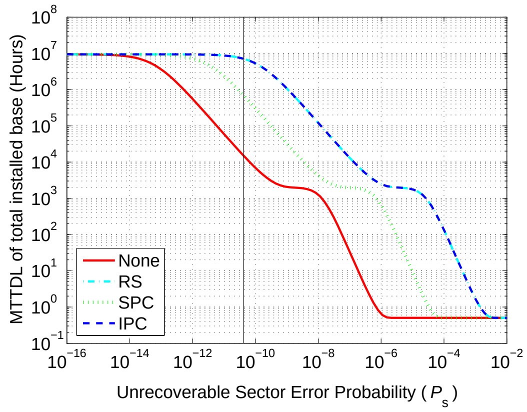

Figure 11: MTTDL for a RAID 6 system with correlated unrecoverable sector errors.

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, performing aggressive read-ahead and write-behind. The response time can be dramatically shortened by increasing the size of 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 simply assume that the RAID controller has a sufficiently large memory so that there are no backpressure effects on the arrival of I/O requests.

The saturation throughput measures the maximum throughput the system can sustain. Here, we do not consider the total response time experienced by the end user or the queue length. We focus our attention solely on the saturation throughput and the disk response time. The saturation throughput is useful for comparing the performance of various RAID schemes. In other words, the higher the saturation throughput, the better the performance of the underlying RAID mechanism. In fact, when the array cache is sufficiently large and the cache replacement strategy is properly tuned, the saturation throughput of RAID scheme is determined by the performance of the disk array, as the bottleneck occurs between the RAID controller and the disk array.

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, the HP Labs' Pantheon for disk arrays [9]. 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 [6], 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 [14] 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 performance of the intra-disk redundancy scheme is practically not affected by the scheduling policy.

We compare the RAID 5 and RAID 6 schemes with the corresponding schemes enhanced by the addition of our proposed 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 utilizes 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. Fig. 12 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 small-write problem compared with RAID 5, suggesting that they are too costly to cope with unrecoverable failures when these are the predominant sources of data loss. In contrast, the RAID 5 and RAID 6 schemes enhanced by the proposed 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 Fig. 13 we investigate the impact of having request sizes that are 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 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 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 [16, 17] that have the largest data records, namely, the Financial 1 (154 MB) and Websearch 2 (139 MB). The traces

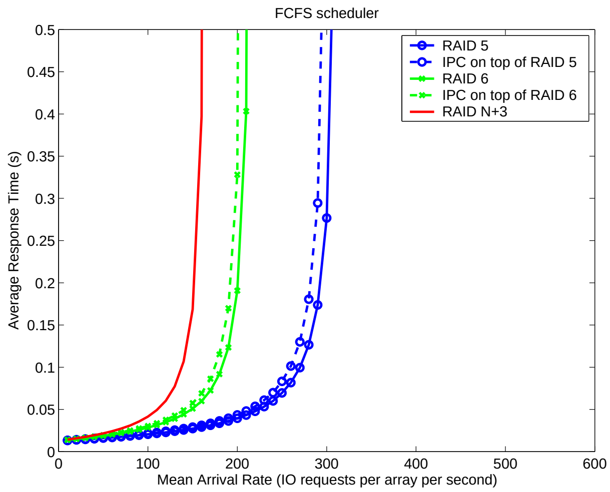

Figure 12: Response time of various RAID systems (synthetic workload, small-write).

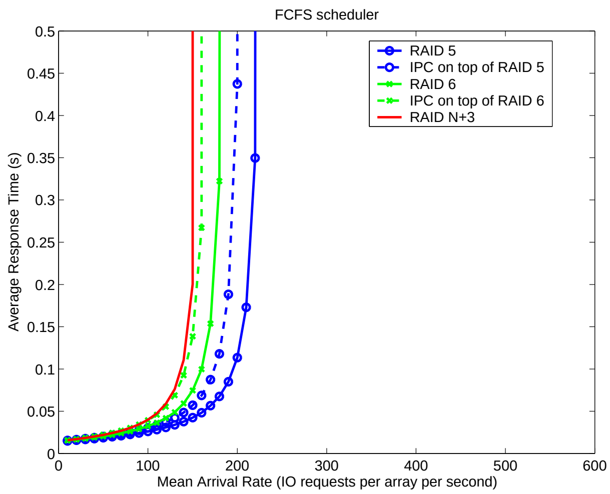

Figure 13: Response time of various RAID systems (synthetic workload, mix of random and sequential requests).

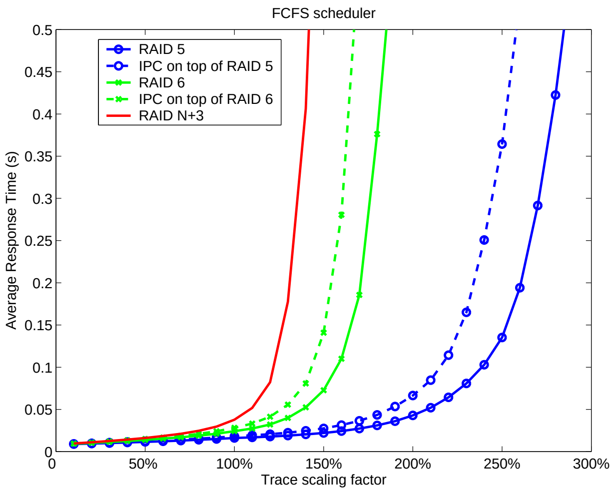

Figure 14: Response time of various RAID systems (SPC Financial 1 trace).

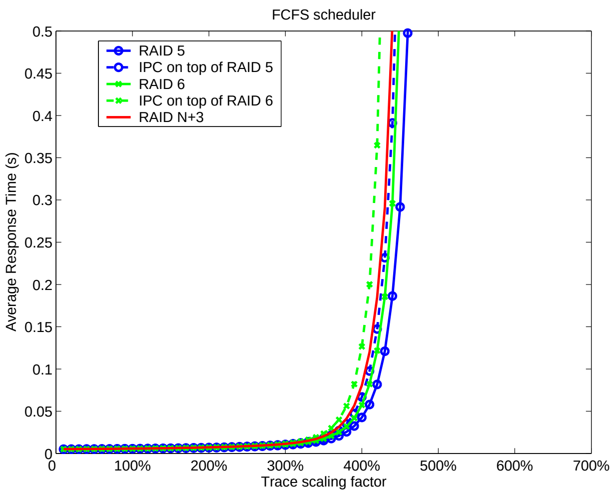

Figure 15: Response time of various RAID systems (SPC Websearch 2 trace).

vary widely in their 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 an identity (100%) scaling factor correspond to the original request stream. Fig. 14 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.

Fig. 15 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 kB to 32 KB. It follows that there is a slight performance hit in the case of the intra-disk redundancy scheme due to alignment issues.

The I/O performance primarily depends on the choice of segment size and to a lesser extent on the data/parity ratio. In general, the larger the segment size, the higher the performance penalty of the write requests. However, for a fixed segment size, the I/O performance penalty decreases slightly as the data/parity ratio increases, but the resistance to media errors is reduced.

7. 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 XOR-based intra-disk redundancy scheme called interleaved parity-check (IPC) coding was introduced and its design described. A new model capturing the effect of correlated unrecoverable sector errors was developed to analyze this scheme. Traditional redundancy schemes based on single-parity-check codes and Reed-Solomon (RS) codes were also analyzed. 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 throughput performance of the enhanced RAID 5 and RAID 6 systems was evaluated by means of simulations. Our results demonstrate that the proposed IPC scheme considerably improves 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, the associated penalty on the I/O performance is minimal. Alternative designs of the intra-disk redundancy concept introduced in this paper, and potential adoption of other erasure coding schemes are subjects of further investigation. Moreover, it would also be interesting to enhance the models presented by taking into account the effects of the caching mechanism.

APPENDIX A
NUMBER OF BURSTS OF ERRORS IN A SEGMENT

Proof of Proposition 1.

Let us consider an instance of k bursts in a segment and let us denote by $\vec{L}$ the vector $(L_1, \ldots, L_k)$ of the corresponding burst lengths and by $\vec{S}$ the vector $(S_1, \ldots, S_k)$ of their corresponding starting sector positions with $1 \le S_1 < \cdots < S_k \le \ell$. 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 $j = 1, 2, \ldots, k-1$. 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 realization in terms of burst lengths $\vec{l} = (l_1, \ldots, l_k)$ and starting sector positions $\vec{s} = (s_1, \ldots, s_k)$. Let us also denote by $\mathcal{R}_k$ the set of all possible realizations $\{(\vec{l}, \vec{s})\}$. 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 size $\hat{B}$, i.e. $P(R_1 = j) = \hat{b}_j$, where $\hat{b}_j \triangleq Pr(\hat{B} = j) = G_j/\bar{B}$ for $j = 1, 2, \ldots$. 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

$j = 1, 2, \ldots, \ell - 1$, 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

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

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

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 size B, i.e. $P(L_k = n) = \sum_{j=n}^{\infty} b_j = G_n$ for $n = 1, 2, \ldots$. In this case it holds that $s_k + l_k = \ell + 1$. Thus,

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

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 the probability of an erroneous sector being the first of its corresponding burst, i.e. $P_{bs} = P_s/\bar{B}$. 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 $P(\vec{L} = \vec{l}, \vec{S} = \vec{s}) = P(I_0 \geq s_1 - 1, \text{burst of errors starts at } s_1, L_1 = l_1, I_1 = s_2 - s_1 - l_1, \ldots, L_k = l_k, I_k \geq \ell + 1 - s_k - l_k) = P(I_0 \geq s_1 - 1)\, P_{bs}\, P(L_1 = l_1)\, P(I_1 = s_2 - s_1 - l_1) \cdots P(L_k = l_k)\, P(I_k \geq \ell + 1 - s_k - l_k) = \alpha^{s_1 - 2}\, \frac{P_s}{\bar{B}}\, b_{l_1}\, (1-\alpha)\alpha^{s_2 - s_1 - l_1 - 1} \cdots b_{l_k}\, \alpha^{\ell - s_k - l_k} = \frac{P_s}{\bar{B}}\, b_{l_1} \cdots b_{l_k}\, \alpha^{\ell - k - (l_1 + \cdots + l_k) - 1}\, (1 - \alpha)^{k-1} = \frac{b_{l_1} \cdots b_{l_k}}{\bar{B}^k}\, P_s^k + O(P_s^{k+1})$.

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

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 $P(\vec{L} = \vec{l}, \vec{S} = \vec{s}) = \frac{A(\vec{l}, \vec{s})}{\bar{B}^k}\, P_s^k + O(P_s^{k+1})$, with $A(\vec{l}, \vec{s})$ being a function of $\vec{l}$, $\vec{s}$ and $\{b_j\}$. Consequently, $P_{\text{burst},k} = \sum_{(\vec{l}, \vec{s}) \in \mathcal{R}_k} P(\vec{L} = \vec{l}, \vec{S} = \vec{s}) = \frac{\sum_{(\vec{l}, \vec{s}) \in \mathcal{R}_k} A(\vec{l}, \vec{s})}{\bar{B}^k}\, P_s^k + O(P_s^{k+1})$. ■

APPENDIX B
REED-SOLOMON (RS) CODING SCHEME

Let us consider an arbitrary segment. 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,

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

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

$$
\begin{aligned}
P_{\text{seg}} &\approx \sum_{l=m+1}^{\ell-1} \frac{G_l}{\bar{B}}\, P_s + \frac{\sum_{j=\ell}^{\infty} G_j}{\bar{B}}\, P_s + \sum_{i=2}^{\ell-m-1} \sum_{l=m+1}^{\ell-i} \frac{b_l}{\bar{B}}\, P_s + \\
&\quad + \sum_{i=2}^{\ell-m} \frac{G_{\ell+1-i}}{\bar{B}}\, P_s = \\
&= \left(\sum_{l=m+1}^{\infty} G_l + \sum_{i=2}^{\ell-m-1} \sum_{l=m+1}^{\ell-i} b_l + \sum_{i=2}^{\ell-m} G_{\ell+1-i} \right) \frac{P_s}{\bar{B}} = \\
&= \left(\sum_{l=1}^{\infty} G_l - \sum_{l=1}^{m} G_l + \sum_{l=m+1}^{\ell-2} \sum_{i=2}^{\ell-l} b_l + \sum_{i=m+1}^{\ell-1} G_i \right) \frac{P_s}{\bar{B}} = \\
&= \left[1 + \frac{(\ell - m - 1) G_{m+1} - \sum_{j=1}^{m} G_j}{\bar{B}} \right] P_s\,.
\end{aligned}
$$

Note that the term in brackets can also be written as $[(\ell - m) G_{m+1} + \sum_{j=m+2}^{\infty} G_j]/\bar{B}$, which is equal to zero iff $G_{m+1} = 0$, i.e. $b_j = 0$ for $j = m + 1, m + 2, \ldots$.

8. REFERENCES

[1] M. Blaum, J. Brady, J. Bruck, and J. Mennon. EVENODD: an efficient scheme for tolerating double disk failures in RAID architectures. *IEEE Trans. Comput.*, 44(2):192–202, Feb. 1995.

[2] W. A. Burkhard and J. Menon. Disk array storage system reliability. In *Proc. Intl. Symp. Fault-Tolerant Comp.*, pages 432–441, June 1993.

[3] P. M. Chen, E.K. Lee, G.A. Gibson, R.H. Katz, and D.A. Patterson. RAID: High-performance, reliable secondary storage. *ACM Computing Surveys*, 26(2):145–185, June 1994.

[4] S. Chen and D. Towsley. A performance evaluation of RAID architectures. *IEEE Trans. Comp.*, 45(10):1116–1130, Oct. 1996.

[5] P. Corbett, R. English, A. Goel, T. Grcanac, S. Kleiman, J. Leong, and S. Sankar. Row-diagonal parity for double disk failure correction. In *Proc. USENIX Conf. File and Storage Technologies*, pages 1–14, 2004.

[6] The DiskSim Simulation Environment (Version 3.0) http://www.pdl.cmu.edu/DiskSim/

[7] G. F. Hughes and J. F. Murray. Reliability and security of RAID storage systems and D2D archives using SATA disk drives. *ACM Trans. Storage*, 1(1):95–107, Dec. 2004.
[8] HDD product information. http://www.hitachigst.com/portal/site/en/
[9] HP, Private Software. http://tesla.hpl.hp.com/private_software/
[10] K. Keeton, C. Santos, D. Beyer, J. Chase, and J. Wilkes. Designing for disasters. In *Proc. USENIX Conf. File and Storage Technologies*, pages 59–72, Mar.-Apr. 2004.
[11] M. Malhotra and K. S. Trivedi. Reliability analysis of redundant arrays of inexpensive disks. *J. Parallel Distrib. Comp.*, 17:146–151, 1993.
[12] M. Malhotra and K. S. Trivedi. Data integrity analysis of disk array systems with analytic modeling of coverage. *Performance Evaluation*, 22:111–133, 1995.
[13] D. A. Patterson, G. Gibson, and R. H. Katz. A case for redundant arrays of inexpensive disks (raid). In *Proc. ACM Conf. Mgmt. Data*, pages 109–116, June 1988.
[14] C. Ruemmler and J. Wilkes. An introduction to disk drive modeling. *IEEE Computer*, pages 17–28, Mar. 1994.
[15] M. Schulze, G. Gibson, R. Katz, and D. Patterson. How reliable is a RAID? In *Proc. IEEE Compcon*, pages 118–123, Mar. 1989.
[16] Storage performance council, storage OLTP application I/O traces. `http://prisms.cs.umass.edu/repository/`.
[17] Storage performance council, storage search engine I/O traces. `http://prisms.cs.umass.edu/repository/`.
[18] K. S. Trivedi. *Probabilistic and Statistics with Reliability, Queueing and Computer Science Applications, 2nd Ed.* Wiley, New York, 2002.
[19] E. Varki, A. Merchant, J. Xu, and X. Qiu. Issues and challenges in the performance analysis of real disk arrays. *IEEE Trans. Par. Distr. Syst.*, 15(6):559–574, June 2004.
[20] X. Wu, J. Li, and H. Kameda. Reliability analysis of disk array organizations by considering uncorrectable bit errors. In *Proc. IEEE Symp. Reliable Distributed Systems*, pages 2–9, 1997.
[21] Q. Xin, E. L. Miller, T. Schwarz, D. D. E. Long, S. A. Brandt, and W. Litwin. Reliability mechanisms for very large storage systems. In *Proc. IEEE/NASA Goddard Conf. Mass Storage Systems and Technologies*, pages 146–156, April 2003.