Outshining Mirrors: MTTDL of Fixed-Order SSPiRAL Layouts

Ahmed Amer[†*] Jehan-François Pâris[‡] Thomas Schwarz[§]
Vincent Ciotola[†] James Larkby-Lahet[†*]
[†]*University of Pittsburgh*
[‡]*University of Houston*
[§]*Santa Clara University*

Abstract

We evaluate the reliability of storage system schemes consisting of n data disks and n parity disks where each parity disk contains the exclusive or (XOR) of two of the n data disks. These schemes are instances of the so-called SSPiRAL (Survivable Storage using Parity in Redundant Array Layouts). Even though they offer the simplicity of mirroring and parity schemes, we show that they approach the performance of much more complex schemes based on erasure coding. In particular, we show that a SSPiRAL scheme defined across six disks offers an MTTDL superior that of three pairs of mirrored disks. Our scheme also offers a higher MTTDL than any scheme capable of surviving the loss of two disks.

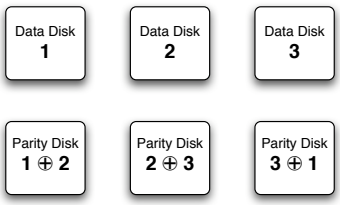
1 Introduction

Complementary trends in hardware and applications are driving an increase in demand for data volume and bandwidth, resulting in an increased risk of data loss and a growing need for improved storage reliability. There is a growing need to survive the failure of multiple storage devices in larger storage arrays, as well as the need to survive the loss of multiple nodes in clustered storage. The volume of digital data is growing, as is the need to build reliable storage infrastrcture. In a recent study, the volume of digital data generated in 2002 was quoted at over 5 Exabytes, 92% of which was written to magnetic disk drives [14, 13]. To put this in perspective, it is equivalent to the 500,000 copies of the contents of the Library of Congress. This estimate does not include the total amount of new digital data transmitted across communication channels (which was on the order of 18 Exabytes). 5 Exabytes represented a doubling of the volume when compared to the figures of three years earlier. And yet, in a more recent study, the 2006 volume of data generated was described as exceeding 160 Exabytes (equivalent to 16,000,000 Libraries of Congress, or twelve stacks of books - each reaching from the earth to the sun) [2]. These numbers are only ballpark figures for our purposes, and yet they clearly demonstrate the rate of data growth, and are particularly daunting when one considers that these figures precede the recent explosion in digital video content online [4]). Such growth will inevitably be reflected in the storage demands of data servers, as well as the storage demands of consumers and producers of such content. This rate of growth is only compounded by the desire - and frequently the need - to retain this data, and will inevitably result in the accelerated growth of the number of data storage devices and servers. More components implies an increased need to protect against the failure of individual components. Data storage devices have a recent history of impressive growth in capacity, this growth alone (assuming it is maintained) could easily be consumed solely by the desire to retain data, and cannot mitigate the increase in storage nodes and devices.

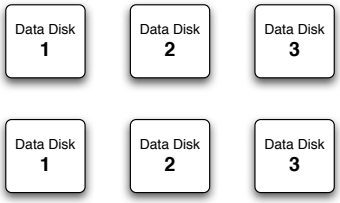
Redundant storage schemes are an obvious solution to increasing reliability, and such applications commonly employ one of two strategies: a combination of replication and parity applied efficiently across an array of devices, or a failure-recovery scheme based on erasure coding. Computational efficiency is important when implementing redundancy schemes for disks, and so parity is particularly appealing due to its ease of computation. There are also combinations of the two approaches, but typically parity schemes tolerate only a small number of component failures,

*Supported in part by the National Science Foundation under Award #0720578.

0-7695-3097-4/08 $25.00 © 2008 IEEE
DOI 10.1109/SNAPI.2007.20



(a) Pairwise-Parity (3+3 SSPiRAL)



(b) 3 pairs of mirrored disks

Figure 1: *Pairwise parity vs. equivalent RAID array.*

while erasure codes tend to be expensive to implement. Excellent parity-based erasure codes and layout schemes have been devised [18, 8], but prior art has focused primarily on aiming to survive a specific number of device failures. We present an argument for an efficient parity-based scheme that compares favorably to erasure codes in terms of reliability, and yet is based on straightforward and efficient parity computations.

2 SSPiRAL Description

SSPiRAL (Survivable Storage using Parity in Redundant Array Layouts) [5] is a redundant data layout scheme based solely on efficient parity computations, offering high reliability and maintainability. Every SSPiRAL layout is defined by three parameters: the degree of the system, the *x-order*, and the total number of nodes available. The degree of a SSPiRAL layout is the number of unique data nodes, while the x-order is the number of nodes that contribute to constructing a parity node. A SSPiRAL arrangement of degree 3 and x-order 2 would use no more than two nodes to build a parity node, and would need a set of six nodes to build a complete layout. Figure 1(a) shows a SSPiRAL layout of degree three and x-order two. Such a layout uses the same number of devices as a mirrored array of three striped disks, as shown in Figure 1(b).

These nodes can be individual devices, servers, or

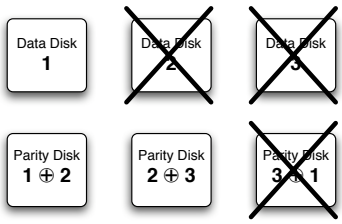


Figure 2: *SSPiRAL data layout and the loss of three nodes.*

storage arrays. SSPiRAL arrangements thereby distinguish between data and parity devices. As long as no devices have failed, the parity updates are efficient to compute, and SSPiRAL has performance comparable to purely striped RAID layouts such as RAID-0 arrays or striped storage clusters such as the original SWIFT distributed storage system [10]. In the example layout of Figure 1(a), data can be written across all three data blocks in parallel, increasing bandwidth, and parity nodes can almost always be calculated without requiring a read from an otherwise busy disk.

An interesting strength of a SSPiRAL layout can be demonstrated through Figure 2, which shows the loss of three of our six devices. In spite of this loss, it is possible to recover all lost data nodes. While a mirrored array *c*an survive the loss of three nodes, there are instances where it cannot survive the loss of two nodes (*e.g.*, it cannot survive the loss of any matched pair of mirrored nodes). There is *no* combination of two node losses that will cause the SSPiRAL layout in Figure 2 to lose data.

3 Reliability Analysis

In this section we evaluate the mean time to data loss (MTTDL) of a SSPiRAL disk array consisting of three data disks and three redundant disks and compare it with the respective MTTDLs of (a) a 3-out-of-6 disk array using an erasure code and (b) an array consisting of three pairs of mirrored disks. All three disk arrays consist of three data disks and three parity disks.

Our system model consists of a disk array with independent failure modes for each disk. When a disk fails, a repair process is immediately initiated for that disk. Should several disks fail, the repair process will be performed in parallel on those disks. We assume that disk failures are independent events exponen-

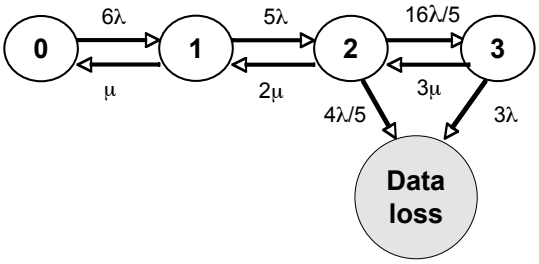


Figure 3: *3+3 disk SSPiRAL array.*

tially distributed with rate λ, and that repairs are exponentially distributed with rate μ.

3.1 3+3 SSPiRAL array

Building an accurate state-transition diagram for a 3+3 SSPiRAL disk array is a task that exceeds the limitations of this paper as we have to distinguish between failures of data disks and failures of parity disks and consider the relations between each data disk and the two parity disks it shares with the two other data disks. Instead, we present here a simplified model.

Observe first that the rate at which an array that has already two failed disks will experience a third disk failure is 4λ. Out of a total of 20 possible outcomes of this failure, only four will cause a data loss. These outcomes are

1. The failure of one data disk and its two parity disks

2. The failure of all three data disks

As a result, we will assume that the rate at which an array that has already two failed disks will incur a disk failure resulting in a data loss will be $4/20 \times 4\lambda = 4\lambda/5$ and the rate at which the same array will incur a disk failure resulting that will not affect the data will be is $16/20 \times 4\lambda = 16\lambda/5$

Figure 3 displays the simplified state transition probability diagram for a 3+3 SSPiRAL array. State $\langle 0 \rangle$ represents the normal state of the array when its six disks are all operational. A failure of any of these disks would bring the array to state $\langle 1 \rangle$. A failure of a second disk would bring the array into state $\langle 2 \rangle$. As we saw before, a failure of a third disk could either result in a data loss or bring the array to state $\langle 3 \rangle$. Any fourth disk failure will result in a data loss.

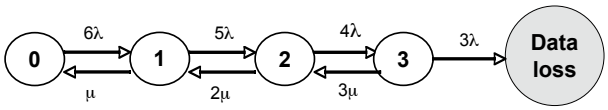


Figure 4: *3-out-of-6 array.*

Repair transitions bring back the array from state $\langle 3 \rangle$ to state $\langle 2 \rangle$, then from state $\langle 2 \rangle$ to state $\langle 1 \rangle$ and, finally, from state $\langle 1 \rangle$ to state $\langle 0 \rangle$. Their rates are equal to the number of failed disks times the disk repair rate μ.

The Kolmogorov system of differential equations describing the behavior of the array is

$$\begin{aligned}
\frac{dp_0(t)}{dt} &= -6\lambda p_0(t) + \mu p_1(t) \\
\frac{dp_1(t)}{dt} &= -(5\lambda + \mu) p_1(t) + 6\lambda p_0(t) + 2\mu p_2(t) \\
\frac{dp_2(t)}{dt} &= -(4\lambda + 2\mu) p_2(t) + 5\lambda p_1(t) + 3\mu p_3(t) \\
\frac{dp_3(t)}{dt} &= -(3\lambda + 3\mu) p_3(t) + \frac{16}{5}\lambda p_2(t)
\end{aligned}$$

where $p_i(t)$ is the probability that the system is in state $\langle i \rangle$ with the initial conditions $p_0(0) = 1$ and $p_i(0) = 0$ for $i \neq 0$.

The Laplace transforms of these equations are

$$\begin{aligned}
sp_0^*(s) &= -6\lambda p_0^*(s) + \mu p_1^*(s) + 1 \\
sp_1^*(s) &= -(5\lambda + \mu) p_1^*(s) + 6\lambda p_0^*(s) + 2\mu p_2^*(s) \\
sp_2^*(s) &= -(4\lambda + 2\mu) p_2^*(s) + 5\lambda p_1^*(s) + 3\mu p_3^*(s) \\
sp_3(s) &= -(3\lambda + 3\mu) p_3^*(s) + \frac{16}{5}\lambda p_2^*(s)
\end{aligned}$$

Observing that the mean time to data loss (MTTDL) of the array is given by

$$MTTDL = \sum_i p_i^*(0),$$

we solve the system of Laplace transforms for $s = 0$ and use this result to obtain the MTTDL of the array:

$$MTTDL = \frac{265\lambda^3 + 137\mu\lambda^2 + 37\mu^2\lambda + 5\mu^3}{60\lambda^3(5\lambda + \mu)}$$

3.2 3-out-of-6 array

Figure 4 displays the state transition probability diagram for a 3-out-of-6 disk array, that is, a disk array

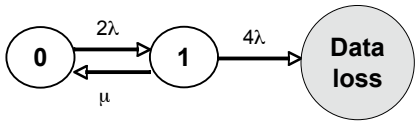


Figure 5: *Single pair of mirrored disks.*

tolerating up to three simultaneous disk failures without data loss. State $\langle 0 \rangle$ represents the normal state of the array when its six disks are all operational. A failure of any of these disks would bring the array to state $\langle 1 \rangle$. A failure of a second disk would bring the array into state $\langle 2 \rangle$ and a failure of a third disk would always bring the array to state $\langle 3 \rangle$. A failure of fourth disk would result in a data loss. Repair transitions are identical to these of a 3+3 SSPiRAL array.

The Kolmogorov system of differential equations describing the behavior of the array is

$$\begin{aligned}
\frac{dp_0(t)}{dt} &= -6\lambda p_0(t) + \mu p_1(t) \\
\frac{dp_1(t)}{dt} &= -(5\lambda + \mu)p_1(t) + 6\lambda p_0(t) + 2\mu p_2(t) \\
\frac{dp_2(t)}{dt} &= -(4\lambda + 2\mu)p_2(t) + 5\lambda p_1(t) + 3\mu p_3(t) \\
\frac{dp_3(t)}{dt} &= -(3\lambda + 3\mu)p_3(t) + 4\lambda p_2(t)
\end{aligned}$$

with the initial conditions $p_0(0) = 1$ and $p_i(0) = 0$ for $i \neq 0$.

Using the same techniques as in the previous case, we obtain the MTTDL of the array:

$$MTTDL = \frac{57\lambda^3 + 23\mu\lambda^2 + 7\mu^2\lambda + \mu^3}{60\lambda^4}$$

3.3 Three pairs of mirrored disks

Figure 5 displays the state transition probability diagram for a single pair of mirrored disks. State $\langle 0 \rangle$ represents the normal state of the array when its two disks are both operational. A failure of either of these disks would bring the array to state $\langle 1 \rangle$ and a failure of a second disk would result in a data loss. The sole repair transition is from state $\langle 1 \rangle$ to state $\langle 0 \rangle$

The two differential equations describing the behavior of the array are

$$\begin{aligned}
\frac{dp_0(t)}{dt} &= -2\lambda p_0(t) + \mu p_1(t) \\
\frac{dp_1(t)}{dt} &= -(\lambda + \mu)p_1(t) + 2\lambda p_0(t)
\end{aligned}$$

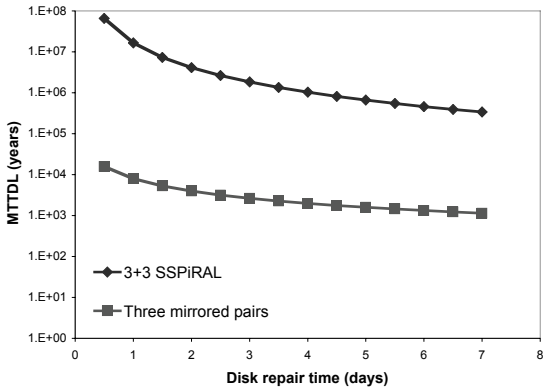


Figure 6: *MTTDL vs mirrored disks.*

with the initial conditions $p_0(0) = 1$ and $p_1(0) = 0$.

Using the same techniques as in the two previous cases, we obtain the MTTDL of the mirrored pair:

$$MTTDL_{pair} = \frac{3\lambda + \mu}{2\lambda^2}$$

The MTTDL of an array consisting of three pairs of mirrored disks is then:

$$MTTDL = \frac{3\lambda + \mu}{6\lambda^2}$$

3.4 Results

Figure 6 displays on a logarithmic scale the MTTDL of SSPiRAL and a mirrored array. We assumed that the disk failure rate λ was one failure every one hundred thousand hours, that is, slightly less than one failure every eleven years. Disk repair times are expressed in days and MTTDLs expressed in years. As we can see, the SSPiRAL disk array provides much better MTTDLs than the array consisting of three pairs of mirrored disks. Both schemes offer identical space utilization, but the SSPiRAL scheme is capable of surviving all two-disk failure scenarios, unlike the mirrored arrays which cannot survive the loss of a matched pair of disks.

In Figure 7 we compare the 3+3 SSPiRAL arrangement to the optimal 3-out-of-6 scheme and the less resilient 4-out-of-6 scheme. The former does not suffer data loss as long as a minimum of three disks are available, while the latter survives the loss of any two disks (requiring four out of six disks to remain available). While it is not surprising that the 3+3 SSPi-

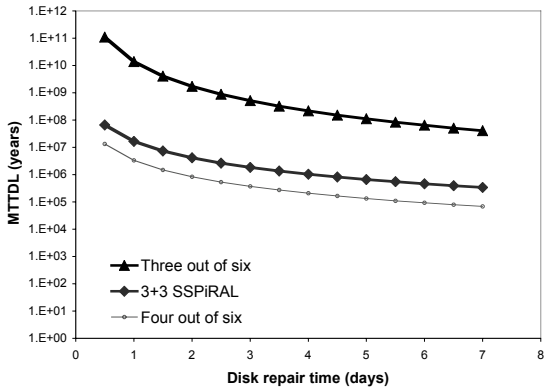


Figure 7: *MTTDL vs optimal erasure coding.*

RAL arrangement falls below the 3-out-of-6 scheme, it is interesting to note it offers a higher MTTDL than the optimal 4-out-of-6 scheme. This is due to the 3+3 SSPiRAL arrangement's ability to survive the failure of three disks. While it cannot survive all such failures, it is nonetheless an improvement upon the 4-out-of-6 erasure coding.

In summary, a simple pairwise SSPiRAL scheme, defined across six disks offers higher MTTDLs than mirroring three disks. It does so while requiring the same 50% space efficiency and the additional effort of inexpensive pairwise parity computations. The parity computations for a SSPiRAL array involve data for only two disks, and are a simple block-wise XOR, which is much more efficient than an optimal erasure coding. The most efficient erasure coding to survive two disk failures offers an improvement in space efficiency, but at a much higher computational cost and a lower reliability than the 3+3 SSPiRAL arrangement.

4 Related Work

Like most of the original RAID layouts [7, 15], SSPiRAL is based solely on parity computations, and like more recent efforts [1, 9, 3, 6] SSPiRAL aims to survive the failure of multiple disks, and to achieve this goal efficiently. SSPiRAL diverges from prior efforts in its definition of efficiency. Unlike row-diagonal parity [6], SSPiRAL does not pursue the goal of optimizing capacity usage, and yet maintains the goals of optimal computational overhead and ease of management and extensibility. SSPiRAL replaces the goal of surviving a *specific* number of disk failures with the goal of surviving the most disk failures possible within the given resource constraints. The basic SSPiRAL layout discussed above can be described as an application of Systematic codes [16] across distinct storage devices. Similarly, such basic SSPiRAL layouts, in their limiting of the number of data sources, are similar to the fixed *in-degree* and *out-degree* parameters in Weaver codes [8] and the earlier $\hat{B}$ layouts [18]. Weaver and $\hat{B}$ are the most similar schemes to SSPiRAL, and all are parity-based schemes using principles first applied in erasure codes for communications applications such as the Luby LT codes, and the later Tornado and Raptor variants [17, 12, 11]. These codes all belong to the class of erasure codes known as low-density parity-check (LDPC) codes. They distinguish themselves from earlier Reed-Solomon and IDA codes by being more efficient to compute at the expense of space utilization. SSPiRAL differs from these prior applications of erasure codes in two major respects: it promises to be more efficient to maintain, and it is implemented with a direct consideration of available system resources, and departing from the requirement to tolerate only a fixed number of device failures.

5 Conclusions & Future Work

The analytical results we present in this paper demonstrate how a basic SSPiRAL array defined across six disks, and using simple pairwise parity, achieves an MTTDL superior to the mirroring of pairs of disks. This SSPiRAL layout offers lower MTTDLs than a complete three-out-of-six erasure code, but depends solely on the simplest pairwise parity computations, and still manages to offer a higher MTTDL than any scheme capable of surviving the loss of two data disks. SSPiRAL arrays are defined based on the degree of parity and the required space efficiency. In this work we have presented a SSPiRAL arrangement that only makes use of pairwise parity operations (thereby limiting the required interconnection bandwidth), and 50% space efficiency (allowing a fair comparison to mirroring schemes). They can be extended to larger numbers of disks and higher parity degrees, offering a trade-off of bandwidth and computational demands against reliability and space efficiency. We plan to investigate the extent and effectiveness of such tradeoffs.

References

[1] G. A. Alvarez, W. A. Burkhard, and F. Cristian, "Tolerating multiple failures in RAID architectures with optimal storage and uniform declustering," in *Proceedings of the 24th International Symposium on Computer Architecture (ISCA)*, (Denver, CO, USA), pp. 62–72, ACM, 1997.

[2] B. Bergstein, "So much data, relatively little space," *Businessweek.com*, Mar. 2007.

[3] M. Blaum, J. Brady, J. Bruck, and J. Menon, "Evenodd: An efficient scheme for tolerating double disk failures in raid architectures," *IEEE Transactions on Computers*, vol. 44, no. 2, pp. 192–202, 1995.

[4] M. Burke, "Ellacoya data shows web traffic overtakes peer-to-peer (P2P) as largest percentage of bandwidth on the network," June 2007.

[5] V. Ciotola, J. Larkby-Lahet, and A. Amer, "SSPiRAL layouts: Practical extreme reliability," Tech. Rep. TR-07-149, Department of Computer Science, University of Pittsburgh, 2007. Presented at the Usenix Annual Technical Conference 2007 poster session.

[6] P. Corbett, B. English, A. Goel, T. Grcanac, S. Kleiman, J. Leong, and S. Sankar, "Row-diagonal parity for double disk failure correction," in *Proceedings of the USENIX Conference on File and Storage Technologies (FAST)*, (San Francisco, CA, USA), pp. 1–14, USENIX Association, 2004.

[7] G. A. Gibson, *Redundant Disk Arrays: Reliable, Parallel Secondary Storage*. PhD thesis, University of California at Berkeley, 1990.

[8] J. L. Hafner, "Weaver codes: Highly fault tolerant erasure codes for storage systems," in *Proceedings of the USENIX Conference on File and Storage Technologies (FAST)*, (San Francisco, CA, USA), Dec. 2005.

[9] K. Hwang, H. Jin, and R. Ho, "RAID-x: A new distributed disk array for I/O-centric cluster computing," in *Proceedings of the 9th IEEE Internation High Performance Distributed Computing Symposium (HPDC)*, pp. 279–286, 2000.

[10] D. D. E. Long, B. R. Montague, and L.-F. Cabrera, "Swift/RAID: A distributed RAID system," *Computing Systems*, vol. 7, no. 3, pp. 333–359, 1994.

[11] M. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman, "Efficient erasure correcting codes.," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 569–584, 2001.

[12] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, D. A. Spielman, and V. Stemann, "Practical loss-resilient codes," in *Proceedings of the 29th ACM Symposium on Theory of Computing (STOC)*, (New York, NY, USA), pp. 150–159, ACM Press, 1997.

[13] P. Lyman and H. R. Varian, "How much storage is enough?," *ACM Queue*, vol. 4, June 2003.

[14] P. Lyman and H. R. Varian, "How much information?," Mar. 2007. http://www.sims.berkeley.edu/how-much-info-2003.

[15] D. A. Patterson, G. Gibson, and R. H. Katz, "A case for redundant arrays of inexpensive disks (RAID)," in *Proceedings of SIGMOD*, pp. 109–116, ACM, 1988.

[16] J. S. Plank and M. G. Thomason, "A practical analysis of low-density parity-check erasure codes for wide-area storage applications," in *Proceedings of the 38th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, (Florence, Italy), June 2004.

[17] A. Shokrollahi, "Raptor codes," *IEEE/ACM Transactions on Networking*, vol. 14, no. SI, pp. 2551–2567, 2006.

[18] B. T. Theodorides and W. A. Burkhard, "B̂: Disk array data layout tolerating multiple failures," in *Proceedings of the IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, (Monterey, CA, USA), pp. 21–32, 2006.