Progressive Parity-Based Hardening of Data Stores

Ahmed Amer* Jehan-François Pâris‡ Darrell D. E. Long† Thomas Schwarz§
*University of Pittsburgh ‡University of Houston
†University of California, Santa Cruz §Santa Clara University

Abstract

We propose the use of parity-based redundant data layouts of increasing reliability as a means to progressively harden data archives. We evaluate the reliability of two such layouts and demonstrate how moving to layouts of higher parity degree offers a mechanism to progressively and dramatically increase the reliability of a multi-device data store. Specifically we propose that a data archive can be migrated to progressively more reliable layouts as the data ages, trading limited (and likely unrealized) increases in update costs for increased reliability. Our parity-based schemes are drawn from SSPiRAL (Survivable Storage using Parity in Redundant Array Layouts) that offer capacity efficiency equivalent to a straightforward mirroring arrangement. Our analysis shows our proposed schemes would utilize no additional physical resources and result in improvements to mean time to data loss of four to seven orders of magnitude.

1 Introduction

With growing volumes of digital data, being retained for longer and longer periods, it is increasingly important to ensure the error-free storage and retrieval of data of varying age from infrastructure that remains physically unchanged. We propose a novel mechanism for increasing the reliability of a data store as the data (and likely the hardware on which it is stored) ages. A simple mirroring of active data can be converted into more reliable and elaborate parity-based redundancy schemes. The potential costs of these schemes are complementary to the decrease in update frequency that is typical of archival data. By the novel use of dedicated storage nodes we demonstrate how any such impact can be varied by selecting among different variants of our redundancy scheme. The two schemes we demonstrate are drawn from a family of redundant layouts we have previously dubbed SSPiRAL (Secure Storage through Parity in Redundant Array Layouts). In Section 2 we describe SSPiRAL layouts and introduce our approach and how it would be applied to ensure increased degrees of reliability as a data archive aged. In Section 3 we present our analysis and analytical results. We discuss related works and conclusions in Sections 4 and 5.

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

†Supported in part by the Petascale Data Storage Institute under Department of Energy award FC02-06ER25768 and by the industrial sponsors of the Storage Systems Research Center.

2 SSPiRAL Description

SSPiRAL (Survivable Storage using Parity in Redundant Array Layouts) [3] are redundant data layout schemes that explicitly consider the possibility that individual disks are not necessarily equally likely to fail. This is achieved by first considering every disk as an independent entity and avoiding the declustering of data across disks, which in turn allows us to survive more failure scenarios than traditional RAID and mirroring schemes. Another side-effect of this approach is that the survival of individual disks may be more critical than others. SSPiRAL layouts can be selected to provide different levels of reliability at the expense of higher update costs, and as such can be well suited to archival applications where the changes in access behavior and reliability demands complement the relative merits and costs of the different SSPiRAL layouts.

2.1 Basic SSPiRAL Layouts

Every SSPiRAL layout is defined by three parameters: the number of data disks, the total number of devices available, and the degree of the system which represents the number of devices that contribute to an individual parity calculation. The number of data disks in a SSPiRAL layout is the number of distinct independent storage devices, representing how many volumes the data will be distributed across (*e.g.* for load balancing or to exploit parallelism). A SSPiRAL arrangement that uses a fixed

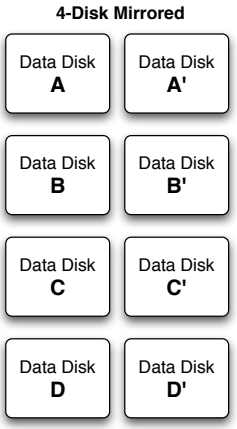

Figure 1: *4 Mirrored Disks, can survive all single and a number of multiple disk failures. It does not survive all double disk failures.*

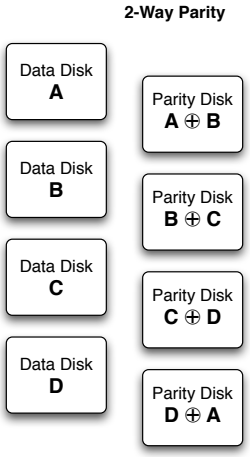

Figure 2: *2-way parity employed with each extra disk, a scheme that now survives all double-disk failures and a number of triple-disk failures.*

degree x is described as a fixed-order array, and in such a layout each parity device holds data that is computed as the result of an XOR operation across exactly x data devices. In this work we have considered only fixed-order layouts.

To specify a SSPiRAL layout, we start with the desired number of data disks in the layout. This is effectively the number of devices that contribute to the data storage capacity of the overall system. From this point, we can impose a constraint on the maximum effort and bandwidth required for parity calculations (by setting the degree, x, of the system) or by setting a limit on the amount of redundant storage we wish to contribute to reducing the likelihood of data loss (by setting the number of additional devices that will hold parity). These parameters are related, as a decision to contribute only a single parity device would necessitate the participation of all data devices in the parity computed at that node. This is effectively a simple $N+1$ RAID scheme with a single parity disk. Adding more parity nodes would give us the freedom to choose between a maximum degree that is equivalent to the number of data devices in the system or a smaller value. Conversely, had we started with a fixed degree, this would have imposed a restriction on the minimum number of parity devices to build a complete SSPiRAL layout. For example, a SSPiRAL arrangement with four data disks and a degree of two would use no more than two data disks to populate a parity disk, and would need a set of eight disks to build a complete layout. Figure 1 shows a mirrored layout with four data disks, while Figure 2 shows a SSPiRAL layout of degree two that also uses eight disks.

2.2 SSPiRAL Layouts for Archival Storage

For archival systems it is important to maintain data reliably and indefinitely. Simultaneously it is true that we are generating more digital data, and a larger percentage of it resides on magnetic disks than ever before. This state makes it desirable to employ redundancy schemes that offer the highest possible storage reliability, whilst avoiding unnecessary additional resource demands. With SSPiRAL we are able to demonstrate how the same number of disks can be dynamically modified, by changing the data layout, to progressively increase the reliability of the overall storage system.

SSPiRAL layouts offer greater reliability than more traditional techniques. In Figure 1 we see a mirrored layout, which is simply a convenient replication of the entire data store, creating an additional copy of each individual disk. This arrangement ensures the survival of the data in the event of the failure of any individual device, and will also survive the loss of any case where any two or three devices carrying different data are lost. The loss of any four devices will definitely result in data loss. On the other hand, in Figure 2 we see one of the simplest SSPiRAL layouts, equivalent to the mirrored layout in Figure 1 in terms of space efficiency and capacity. The difference lies in its use of pairwise parity disks in place of the straighforward mirrors of individual disks. In this arrangement the same number of primary (required for data) and redundant (additional) disks offer the ability to survive all two-disk failures and several three disk failures as well. This is achieved at the expense of requiring parity-updates when data is modified, but the result is redundant disks that are

useful to more than one primary disk, and ultimately more ways to reproduce data in the event of a disk failure.

SSPiRAL layouts, when employed without declustering of data and parity across disks, offer advantages of mirroring while exceeding the reliability of other RAID-like approaches. In addition to being simple to implement and intuitive, the basic mirroring scheme of Figure 1 offers the ability to exploit knowledge of device interdependencies. For example, it would improve the effective reliability of the storage system if we ensure a physical separation between each device and its mirror. Using separate physical housings for such pairs, whose correlated failures guarantee data loss, avoids tragedy in the event of physical damage to a housing holding a set of disks. This ability to group disks into correlated groups is typically lost when we employ more traditional parity-based schemes, which employ declustering to improve recovery times and update performance. In Figure 3 we see how the contents of four disks can be physically distributed amongst four disks while maintaining the same logical structure. This declustering has several benefits, most notably offering a way to exploit parallelism by avoiding bottlenecks and also reducing the time to restore the contents of a lost device. For example, if $D = A \oplus B \oplus C$, then the reconstruction of a lost disk could be performed with the parallel cooperation of all surviving disks. Unfortunately, this interdependence also results in a reduction of reliability when not dealing with a simple $N+1$ parity scheme. It is easy to see this with mirroring as an example. Assuming $A = C$ and $B = D$ (the last two disks mirror the contents of the first two), then it becomes clear that the loss of any two disks would result in some data loss. This example is particularly simplistic, and we would not suggest that declustering would be applied to such a configuration, we use it merely to illustrate the importance of dedicated roles for dedicated volumes (whether they be individual disks or subsystems) in a SSPiRAL layout.

SSPiRAL layouts can be progressively modified to suit an aging data store. While SSPiRAL layouts of higher degree incur potentially higher update costs than simple replication we believe them to be more reliable. We should note that employing our SSPiRAL layouts incurs no penalties for read requests, and the penalties for write requests are limited to the need to update one or two additional devices. Such write traffic involves the use of a strictly limited number of additional devices, can likely be delayed and batched for improved efficiency, and is ultimately less likely to be encountered as data ages. We would further argue that far from a penalty, there are potential performance savings in device energy consumption and load balancing thanks to the increased number of

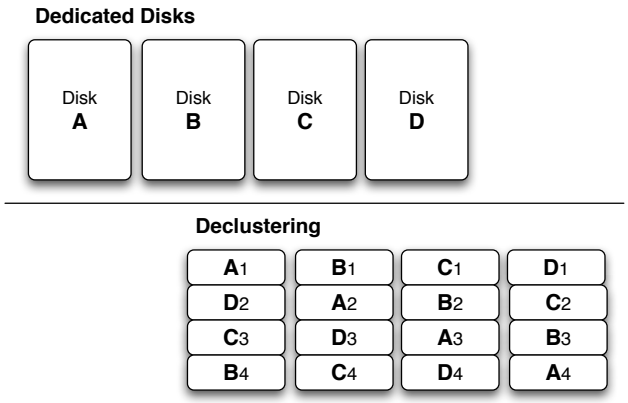

Figure 3: *Declustering data across devices aids in reducing recovery time by allowing parallelism and slightly reducing the volume of data to be restored. It also results in a failure model that guarantees data loss in the event of the failure of the minimum number of devices that* could *result in data loss.*

.

paths for recovering a specific data block. Energy and response time benefits are heavily workload dependent and evaluating them is part of our future work. Here our hypothesis is that employing SSPiRAL layouts of progressively higher degree offers both the opportunity to refresh stored data and the increased reliability demanded by long-term archival on magnetic disks. Converting a data store from one layout to another can be done gradually, by applying the new layout to a stripe of blocks across all participating disks, until the entire data store has been converted to the new layout. Such a process will require a reading of the entire data set, which can serve as the data scrubbing process to refresh archived data. The potential performance cost of employing progressively more reliable SSPiRAL layouts is further mitigated and avoided thanks to the nature of archival data, which becomes more referential and experiences fewer updates as it ages and reaches a more quiescent state. The usefulness of starting with a mirrored layout, and transitioning to increasingly reliable SSPiRAL layouts is dependent on the workload being suited to such a transition, and most importantly on the validity of our hypothesis that increasing the degree of a SSPiRAL layout significantly improves reliability. We validate this hypothesis in the following section.

3 Reliability Analysis

We now evaluate the mean time to data loss (MTTDL) of SSPiRAL disk arrays consisting of four data disks and three to four 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 exponentially distributed with rate λ, and that repairs are exponentially distributed with rate μ.

3.1 4+4 SSPiRAL array of degree 2

Building an accurate state-transition diagram for a 4+4 SSPiRAL disk array of degree 2 is beyond the scope of this work as we would 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 three other data disks. Instead, we present here a simplified model.

Figure 2 displays a 4+4 SSPiRAL array of degree 2 with four data disks respectively labeled A, B, C and D and four parity disks respectively labeled AB, BC, CD and DA such that $AB = A \oplus B$, $BC = B \oplus C$ and so forth.

Observe first that the array can tolerate the simultaneous failures of two arbitrary disks but not all triple disk failures since the failure of one of the four data disks and its two parity disks—say data disk A and parity disks AB and DA—will result in the irrecoverable loss of the data stored in that disk. This is to say that 4 triple failures out of a total of 56 will be fatal. Since the rate at which an array that has already two failed disks will experience a third disk failure is 6λ, we will assume that the rate at which an array that has already two failed disks will incur a data loss will be $4/56 \times 6\lambda = 3\lambda/7$ and the rate at which the same array will incur a disk failure that will not affect the data will be $52/56 \times 6\lambda = 39\lambda/7$.

In addition, we will assume that all quadruple disk failures will result in a data loss. While this assumption is not strictly true, it will not significantly affect our results as long as quadruple failures remain very infrequent occurrences.

Figure 4 displays the simplified state transition probability diagram for a 4+4 SSPiRAL array of degree 2. State $\langle 0 \rangle$ represents the normal state of the array when its eight 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 $\langle 2 \rangle$. 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 is assumed to result in a data loss.

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.

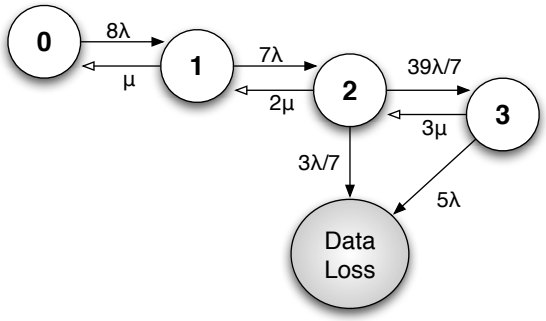

Figure 4: *Markov model for 2-way parity scheme.*

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

$$\begin{aligned}
\frac{dp_0(t)}{dt} &= -8\lambda p_0(t) + \mu p_1(t) \\
\frac{dp_1(t)}{dt} &= -(7\lambda+\mu) p_1(t) + 8\lambda p_0(t) + 2\mu p_2(t) \\
\frac{dp_2(t)}{dt} &= -(6\lambda+2\mu) p_2(t) + 7\lambda p_1(t) + 3\mu p_3(t) \\
\frac{dp_3(t)}{dt} &= -(5\lambda+3\mu) p_3(t) + \frac{39}{7}\lambda p_2(t)
\end{aligned}$$

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

$$\begin{aligned}
s p_0^*(s) &= -8\lambda p_0^*(s) + \mu p_1^*(s) + 1 \\
s p_1^*(s) &= -(7\lambda+\mu) p_1^*(s) + 8\lambda p_0^*(s) + 2\mu p_2^*(s) \\
s p_2^*(s) &= -(6\lambda+2\mu) p_2^*(s) + 7\lambda p_1^*(s) + 3\mu p_3^*(s) \\
s p_3^*(s) &= -(5\lambda+3\mu) p_3^*(s) + \frac{39}{7}\lambda p_2^*(s)
\end{aligned}$$

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:

$$MTTDL = \frac{7294\lambda^3 + 2081\lambda^2\mu + 415\lambda\mu^2 + 42\mu^3}{168\lambda^3(70\lambda + 3\mu)}$$

3.2 4+4 SSPiRAL array of degree 3

Figure 5 displays a 4+4 SSPiRAL array of degree 3 with four data disks respectively labeled A, B, C and D and

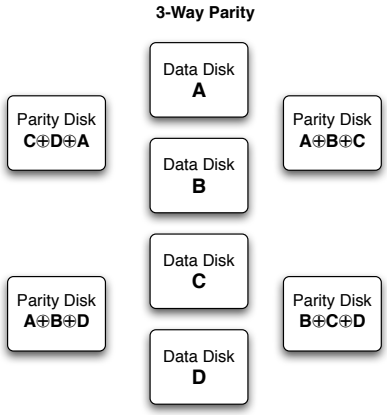

Figure 5: *3-way parity employed with each extra volume, a scheme that now survives all triple-disk failures and a number of quad-disk failures.*

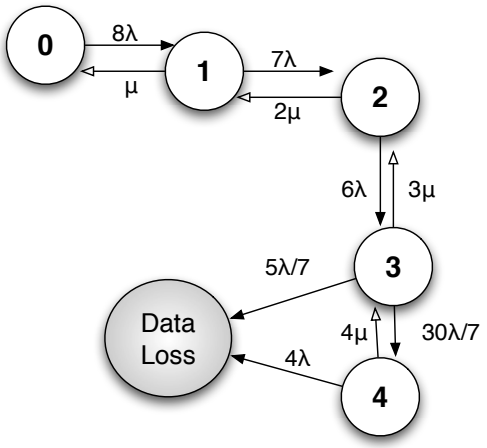

Figure 6: *Markov model for 3-way parity scheme.*

four parity disks respectively labeled *ABC*, *BCD*, *CDA* and *DAB* such that $AB = A \oplus B \oplus C$, $BCD = B \oplus C \oplus D$ and so on.

Observe first that the array can now tolerate the simultaneous failures of up to three arbitrary disks. The sole quadruple failures that will result in a data loss are

- a failure of one of the four data disks and its three parity disks—say, data disk *A* and parity disks *ABC*, *CDA* and *DAB*—or
- a failure of two data disks and the two parity disks not containing their exclusive or—say data disks *A* and *B* and parity disks *BCD* and *CDA*.

There are thus 10 (*i.e.* $4+6$) quadruple failures out of a total of 70 that result in a data loss. Since the rate at which an array that has already three failed disks will experience a fourth disk failure is 5λ, the rate at which an array that has already three failed disks will incur a data loss will be $10/70 \times 5\lambda = 5\lambda/7$ and the rate at which the same array will incur a disk failure that will not affect the data will be $60/70 \times 5\lambda = 30\lambda/7$.

Figure 6 displays the simplified state transition probability diagram for a 4+4 SSPiRAL array of degree 3. State $\langle 0 \rangle$ represents the normal state of the array when its eight 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 $\langle 2 \rangle$. A failure of a third disk would bring the array into state $\langle 3 \rangle$. A failure of a fourth disk could either result in a data loss or bring the array to state $\langle 3 \rangle$. Any fifth disk failure would result in a data loss.

Repair transitions bring back the array first from from state $\langle 4 \rangle$ to state $\langle 3 \rangle$, then from from state $\langle 3 \rangle$ to state $\langle 2 \rangle$, and so forth until the array returns to its original 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

$$\begin{aligned}
\frac{dp_0(t)}{dt} &= -8\lambda p_0(t) + \mu p_1(t) \\
\frac{dp_1(t)}{dt} &= -(7\lambda+\mu)p_1(t) + 8\lambda p_0(t) + 2\mu p_2(t) \\
\frac{dp_2(t)}{dt} &= -(6\lambda+2\mu)p_2(t) + 7\lambda p_1(t) + 3\mu p_3(t) \\
\frac{dp_3(t)}{dt} &= -(5\lambda+3\mu)p_3(t) + 6\lambda p_2(t) + 4\mu p_4(t) \\
\frac{dp_4(t)}{dt} &= -(4\lambda+4\mu)p_4(t) + \frac{30}{7}\lambda p_3(t)
\end{aligned}$$

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 $i \neq 0$.

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

$$MTTDL = \frac{4991\lambda^4 + 2514\lambda^3\mu + 846\lambda^2\mu^2 + 194\lambda\mu^3 + 21\mu^4}{840\lambda^4(7\lambda+\mu)}$$

3.3 4+3 SSPiRAL array of degree 3

Let us now consider what happens if one of the four parity disks of a 4+4 SSPiRAL array of degree 3 is removed either as the result of a failure or to allow it to be used in some other capacity. Figure 7 displays such an array. It was derived from the 4+4 SSPiRAL array of Figure 5 by removing parity disk *CDA*.

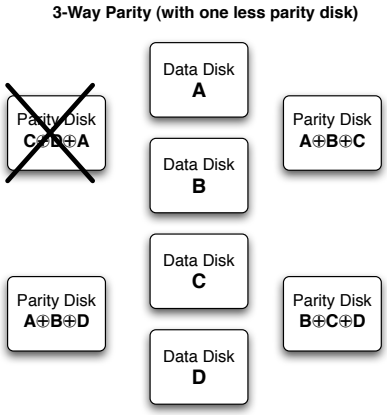

Figure 7: *A degraded 3-way parity scheme, utilizing only three additional disks. This scheme can no longer survive all triple-disk failures, but continues to offer excellent failure tolerance.*

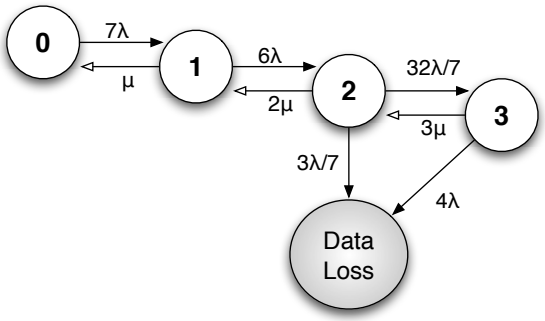

Figure 8: *Markov model for 3-way parity scheme, but with one missing parity disk.*

Observe first that the design is asymmetrical and does not afford equal protection to the contents of all four data disks. While the three parity disks provide three ways to reconstruct the contents of data disk *B*, they only provide two ways to reconstruct the contents of the three other data disks.

As a result, any triple failure involving one of these three data disks and its two parity disks—say, data disk *A* and parity disks *ABC* and *DAB*—will result in a data loss. The 32 other triple failures out of a total of 35 will not affect the data. Observing that the rate at which an array that has already two failed disks will experience a third disk failure is 5λ, we see that the rate at which an array that has already two failed disks will incur a data loss will be $3/35 \times 5\lambda = 3\lambda/7$ while the rate at which the same array will incur a disk failure that will not affect the data will be $32/35 \times 5\lambda = 32\lambda/7$. All quadruple disk failures will result in a data loss.

Figure 8 displays the simplified state transition probability diagram for a 4+3 SSPiRAL array of degree 3. State $\langle 0 \rangle$ represents the normal state of the array when its eight 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 $\langle 2 \rangle$. A failure of a third disk could either result in a data loss or bring the array to state $\langle 3 \rangle$.

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

$$\begin{aligned}
\frac{dp_0(t)}{dt} &= -7\lambda p_0(t) + \mu p_1(t) \\
\frac{dp_1(t)}{dt} &= -(6\lambda+\mu)p_1(t) + 7\lambda p_0(t) + 2\mu p_2(t) \\
\frac{dp_2(t)}{dt} &= -(5\lambda+2\mu)p_2(t) + 6\lambda p_1(t) + 3\mu p_3(t) \\
\frac{dp_3(t)}{dt} &= -(4\lambda+3\mu)p_3(t) + \frac{32}{7}\lambda p_2(t)
\end{aligned}$$

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 $i \neq 0$.

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

$$MTTDL = \frac{4340\lambda^3 + 1531\lambda^2\mu + 359\lambda\mu^2 + 42\mu^3}{42\lambda^3(140\lambda + 9\mu)}$$

3.4 Comparative Results

From our analysis and derived MTTDL expressions we are able to compare the expected MTTDL of SSPiRAL layouts of degree two and three to a basic mirroring scheme. The results are for scenarios with four data disks, and clearly demonstrate a dramatic improvement in reliability as the degree is increased. Figure 9 plots the MTTDL against a range of average repair times from twelve hours to seven days. This is a reasonable range of repair rates given current disk capacities and the expectation that a replacement for a failed disk may be on-hand or require a delivery period of several days, as such a repair would need to include time to restore data to the new disk. Switching to pairwise parity in place of disk replicas results in an MTTDL improvement of four orders of magnitude, jumping from a maximum of roughly ten thousand years to a maximum of a hundred million years. This performance improvement is repeated for three-way parity as

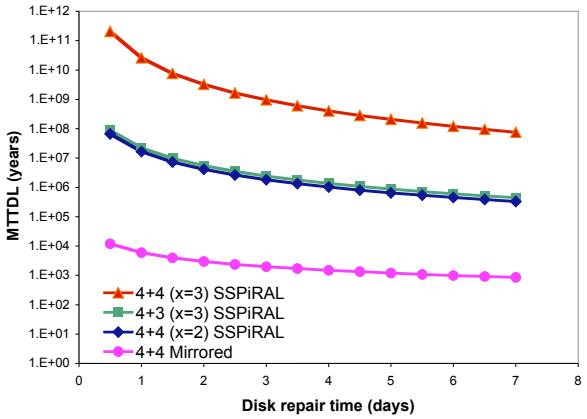

Figure 9: *MTTDL comparing mirrored, two-way parity and three-way parity schemes. Of particular note is increased reliability of the three-way parity schemes when faced with the loss of one parity disk.*

we observe a jump to a maximum of well over ten billion years. While we do not expect a physical data store to remain unchanged for such lengths of time, we present these MTTDLs as strong indicators of the relative reliability of these schemes. It is particularly interesting to note that the loss of an individual parity disk in the SSPiRAL layout with a degree of three results in an MTTDL slightly superior to that of the SSPiRAL layout with degree two and no failed disks.

4 Related Work

SSPiRAL layouts were first introduced as simple parity schemes that could potentially be tailored to requirements and disk availability of a storage system [3, 5, 2], but benefits of dynamically modifying a layout were explored in even earlier work [14]. While the use of dedicated volumes proposed with SSPiRAL schemes is novel, more recent efforts have started to investigate broader families of parity-based codes applied to heterogenous device [8]. Like most of the original RAID layouts [7, 15], SSPiRAL is based solely on parity computations, and like more recent efforts [1, 10, 4, 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 [9] 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, 13, 12]. 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

We have presented an argument for the use of parity based redundancy schemes for progressively hardening data stores as they age. Through analysis of the reliability of SSPiRAL schemes with increasing degree, we have demonstrated how such schemes can offer reliability improvements of four to three orders of magnitude while using the same number of active and backup devices as an initial simple mirroring scheme. This comes at a potential increase in the performance impact of updates to stored data, but such a cost is typically unrealized if the data is updated less frequently as it ages. This approach is particularly suited to archive data that grows less active and undergoes a transition from a heavily accessed and updated data set, to a more referential and stable one. This is a typical expectation for archival data, and is true of many data storage applications. Our scheme is novel in both the general approach and the emphasis on the use of dedicated (or at least non-overlapping) roles for individual volumes. This approach, which is contrary to the conventional tendency to decluster parity and data in a redundancy scheme, allows for the ability to adjust the layout to the current workload and risk of data loss. While we have shown that schemes involving a higher parity degree offer a greater resilience to individual volume failures, we have further demonstrated that a three-way parity scheme continues to offer exceptional reliability even while operating with a failed parity disk.

The specific use of parity-based schemes such as SSPiRAL can be tailored to the status of the data store. Transitioning from one layout to another can be put into effect gradually, and combined with the process of scrubbing data on magnetic disks. Simple device-wise mirroring can progress to parity-based layouts of higher reliability as the data ages and becomes less frequently updated. This allows the system to survive a greater number of disk failures as it ages and the likelihood of such failures increases. This is effectively a hardening of a data store as the data itself becomes more long-lived, as opposed to data that is effectively destroyed and replaced with frequent updates. The former (long-lived, archival, infrequently updated data) is preferred for the more complex SSPiRAL layouts and is in greatest need of the most reliable layouts.

Our proposed system deliberately avoids declustering across devices, and this contributes greatly to its impressive reliability results. The example of the SSPiRAL layout of degree three, which offers higher reliability with seven disks than an eight-disk SSPIRAL layout utilizing pairwise parity (degree two), would not have been possible had the contents of the disks been declustered. While declustering is an important tool, it need not be applied at the level of independent devices. For example, individual "disks" in SSPiRAL may in fact be volumes formed of multiple disks in RAID arrays. In such a case, the individual arrays would employ declustering intra-volume, while our scheme would dictate the dedicated roles for each volume. The use of dedicated disks in our proposed scheme allowed us to exploit the likelihood of failures of distinct individual disks. Our analysis, however, made the conservative assumption that all disks were equally likely to fail, but in practice we can potentially do better. Assigning the parity disks to the oldest (or newest) hardware increases the likelihood that a failure will occur on a benign volume that results in the least impact on system performance. The impact of heterogeneous devices and exploiting knowledge of disparate failure likelihoods is a subject of future work.

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] A. Amer, D. D. Long, J.-F. Paris, and T. Schwarz, "Increased reliability with SSPiRAL data layouts," in *Proceedings of the IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, (Baltimore, MD, USA), 2008.

[3] A. Amer, J.-F. Paris, T. Schwarz, V. Ciotola, and J. Larkby-Lahet, "Outshining Mirrors: MTTDL of Fixed-Order SSPiRAL Layouts," in *Proceedings of the International Workshop on Storage Net- work Architecture and Parallel I/Os (SNAPI07)*, (San Diego, CA, USA), Sept. 2007.

[4] 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.

[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] K. M. Greenan, E. L. Miller, and J. J. Wylie, "Reliability of XOR-based erasure codes on heterogeneous devices," in *Proceedings of the Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, 2008.

[9] 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.

[10] 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.

[11] 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.

[12] 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.

[13] 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.

[14] J.-F. Pâris, T. J. E. Schwarz, and D. D. E. Long, "Self-adaptive disk arrays," in *Proceedings of the 8th International Symposium on Stabilization, Safety, and Security of Distributed Systems*, (Dallas, TX, USA), pp. 469–483, Nov. 2006.

[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.