High-fidelity reliability simulation of XOR-based erasure codes (Poster submission / Extended abstract)

Kevin Greenan and Jay J. Wylie
Hewlett-Packard Labs
kmgreen@soe.ucsc.edu, jay.wylie@hp.com

1 Introduction

Erasure codes are the means by which storage systems are typically made reliable. Recent high profile studies of disk failure and sector failures indicate that ever more fault tolerant erasure codes are needed. Many traditional RAID approaches, parity-check array codes (e.g.,EVENODD, RDP, and X-code), and MDS codes offer two and three disk fault tolerant schemes. There are also many novel erasure code proposals that provide similar fault tolerance, such as SPC codes, Weaver codes, and low-density parity-check (LDPC) codes. Such erasure codes offer different space-efficiency and performance tradeoffs than traditional erasure codes. Unfortunately, such erasure codes are also distinguished by their *irregular* fault tolerance: different sets of disk and sector failures of similar sizes may or may not lead to data loss. Some RAID schemes provide irregular fault tolerance. For example, replicated stripes (RAID 10) has pairs of disk failures that lead to data loss, and others that do not.

Reasoning about irregular fault tolerance and reliability is quite challenging. Hafner and Rao have developed Markov models of the reliability of some irregular erasure codes [3]. Elerath and Pecht recently concluded that simulation of a single-disk fault tolerant storage system utilizing Weibull-distributed failure rates and latent sector failures, leads to radically different MTTDL results than Markov models [1]. We go beyond the conclusions of Elerath and Pecht: we believe that Markov models cannot effectively model irregular fault tolerance because of the complexities of correctly modeling disk rebuild, and because latent sector failures and scrubbing should be included in the model [2].

We have developed the High-Fidelity Reliability (HFR) Simulator. The HFR Simulator permits the reliability evaluation of both irregular and traditional erasure codes under a single framework. To achieve *high-fidelity* simulation we leverage our prior work on *minimal erasures* [4]. Minimal erasures concisely and precisely describe the fault tolerance of an irregular erasure code. We used the HFR Simulator to perform the most comprehensive "apples to apples" comparison of the reliability of different erasure codes of which we are aware. We evaluated over ten different erasure code constructions in the same framework. In the comparison, we evaluate maximum distance separable (MDS) codes (i.e., Reed-Solomon, RAID 4, RAID 6), parity-check array codes (i.e., EVENODD, RDP, X-code, SPC, and Weaver codes), and low-density parity-check (LDPC) codes.

2 High-fidelity simulation

We have designed and built the High-Fidelity Reliability (HFR) Simulator, a reliability simulator for erasure-coded storage. The basic simulation method—Monte Carlo discrete event simulation—is itself not novel. Indeed, at a high level, the design of the HFR Simulator has many similarities to the simulator recently described by Elerath and Pecht [1]. The HFR Simulator is *high-fidelity* in that it accurately simulates the reliability of erasure codes with irregular fault tolerance that can tolerate two or more disk failures, with regard to both disk and sector failures.

The HFR Simulator operates as follows. It evaluates the reliability of a single array and is initialized by drawing failure times for each disk in the array. A disk rebuild time is drawn for the disk that fails earliest. Any disk failure that intersects the earliest disk failure and recovery also draws its recovery time. Sector failures, and corresponding scrub times, are drawn for each disk up to the time of the earliest disk recovery. The failure histories are analyzed to identify periods during which there are at least a Hamming distance number of failures within some stripe. Such periods are further analyzed to determine if data is lost. If data is not lost, the disk failure times are drawn for all recovered disks, and the process repeats until a data loss event occurs.

Much of the novelty of the HFR Simulator lies in the methods it uses to efficiently determine if a set of disk and sector failures leads to data loss. There are two methods of such bookkeeping: for highest fidelity, the MEL is used, and for coarser-grained analysis, the *fault tolerance* *matrix* is used. To use the MEL, elements corresponding to disk and sector failures are removed from minimal erasures in the MEL. An empty minimal erasure indicates data loss. Various data structures and bitmap representations are used to efficiently update minimal erasures and to determine if any minimal erasure is empty. The fault tolerance matrix is a precomputed table that lists the probability that a specific number of disk failures and sector failures leads to data loss. Each stripe in the storage array that has some sector failures leads to a separate random draw to determine if data is lost. The fault tolerance matrix is a generalized version of the *conditional probabilities* vector used by Hafner and Rao [3]. The fault tolerance matrix is a coarser-grained method of bookkeeping than the MEL method, and is only suitable for irregular codes that exhibit some symmetry (e.g., because they are rotated).

The HFR Simulator is implemented in Python. As input, it takes descriptions of the erasure code, the storage system (number of disks, size of disks, etc.), and the distributions (disk & sector failure, disk rebuild, and sector scrub). Currently, exponential and Weibull distributions can be specified. The simulator outputs MTTDL and Data Loss Events per Petabyte-Year (DLE per PB-YEAR).

3 Example results

We evaluate the reliability of many XOR-based erasure code constructions in an "apples to apples" comparison of reliability. We use the same disk failure, disk rebuild, sector failure, and sector scrubbing distributions as Elerath and Pecht, including the β and γ values for Weibull distributions (cf. Table 2 in [1]).

Table 1 lists the results of the reliability simulations. For each code, the rate (i.e., space efficiency), Hamming distance, and reliability in Data Loss Events per Petabyte Year (DLE per PB-YEAR) is listed. All of the codes use 8 disks, except the X-code which uses 7 disks. The Hamming distance, d, is one greater than the code's disk fault tolerance. The smaller the DLE per PB-YEAR number, the more reliable the code.

More extensive details on all of the codes evaluated are given in the full version of this paper [2]. The MDS codes are effectively RAID 4 and RAID 6. Three of the parity-check array codes are two disk fault tolerant (EVENODD, RDP, X-code), and the fourth, SPC (simple product code), is single disk-single sector fault tolerant. The FLAT codes are horizontal XOR-based codes found via computational techniques [4]. The RAID 10 code is traditional replicated striping over 8 disks. The Weaver codes are ones published by Hafner.

Other than the MDS codes, all of the erasure codes evaluated exhibit some irregularity. EVENODD, RDP, SPC, and X-code only exhibit irregularity in their sector fault tolerance. Since double sector failures are so rare though, this irregularity is not interesting. The remaining codes exhibit radically different disk and sector fault tolerances because of their distinct irregular constructions.

code	**rate**	d	DLE per PB-YEAR
(6,2)-MDS	0.75	3	0.0014
(7,1)-MDS	0.88	2	5.4488
(36,12)-EVENODD	0.75	3	0.0015
(36,12)-RDP	0.75	3	0.0014
(48,16)-XCODE	0.75	3	0.0011
(42,12)-SPC	0.75	2	0.0211
(5,3)-FLAT	0.62	2	0.3398
(6,2)-FLAT	0.75	2	1.858
(4,4)-RAID 10	0.50	2	3.039
(8,8,1)-WEAVER	0.50	2	0.7567
(8,8,2)-WEAVER	0.50	3	0.0001

Table 1: Reliability of XOR-based erasure codes.

We believe that this "apples to apples" comparison of reliability of different erasure codes is the most extensive to date, and is a significant accomplishment. We are hesitant to draw sweeping conclusions from the results in Table 1 though. Reliability is just one aspect of a storage system. Performance and cost are other key aspects of any comprehensive evaluation. Whereas, "apples to apples" performance and cost analysis of such codes was previously possible, such a reliability comparison was not. The HFR Simulator makes such a comparison possible.

References

[1] J. F. Elerath and M. Pecht. Enhanced reliability modeling of raid storage systems. In *DSN-2007*, pages 175–184. IEEE, June 2007.

[2] K. Greenan and J. J. Wylie. On the reliability of XOR-based erasure codes. Technical report, HP, February 2008.

[3] J. L. Hafner and K. Rao. Notes on reliability models for non-MDS erasure codes. Technical Report RJ–10391, IBM, October 2006.

[4] J. J. Wylie and R. Swaminathan. Determining fault tolerance of XOR-based erasure codes efficiently. In *DSN-2007*, pages 206–215. IEEE, June 2007.