SAND2021-11420C

Characterizing Memory Failures Using Benford's Law

Kurt B. Ferreira & Scott Levy

Center for Computing Research
Sandia National Laboratories
{kbferre, sllevy}@sandia.gov

Abstract. Fault tolerance is a key challenge as high performance computing systems continue to increase component counts, individual component reliability decreases, and hardware and software complexity increases. To better understand the potential impacts of failures on next-generation systems, significant effort has been devoted to collecting, characterizing and analyzing failures on current systems. These studies require large volumes of data and complex analysis in an attempt to identify statistical properties of the failure data.
In this paper, we examine the lifetime of failures on the Cielo supercomputer that was located at Los Alamos National Laboratory, looking specifically at the time between faults on this system. Through this analysis, we show that the time between uncorrectable faults for this system obeys Benford's law, This law applies to a number of naturally occurring collections of numbers and states that the leading digit is more likely to be small, for example a leading digit of 1 is more likely than 9. We also show that a number of common distributions used to model failures also follow this law. This work provides critical analysis on the distribution of times between failures for extreme-scale systems. Specifically, the analysis in this work could be used as a simple form of failure prediction or used for modeling realistic failures.

1 Introduction

Fault tolerance is a key challenge as high performance computing systems continue to increase component counts, individual component reliability decreases, hardware complexity increases, and software complexity increases. To better understand the potential impacts on next-generation systems, significant effort has been devoted to collecting, characterizing and analyzing failures [26, 25, 15, 19, 16]. These studies require large volumes of data, typically gathered over many years, and utilizing complex analysis in an attempt to identify the underlying probability distribution and its statistical properties.

Several mitigation methods have been developed to address memory failures. A popular method of fault tolerance in today's large-scale production systems is coordinated checkpoint/restart. The overheads of checkpoint/restart are determined, in part, by the duration of the checkpoint interval. Determining the

optimal checkpoint interval requires an understanding of failure statistics on a given system in order to minimize lost work and checkpoint overheads [8]. Therefore, to better understand checkpoint overheads, one must understand the failure rate on a system. Checkpointing can also be coupled with failure prediction [14] to minimize time lost in the *rework* stage, but current prediction-based mechanisms have relatively poor performance or exceedingly high overheads. Therefore, having a cheap method to determine when faults are likely could improve application performance.

In this paper we examine faults on the entire lifetime of the Cielo supercomputer that was located at Los Alamos National Laboratory, looking specifically at the time between memory faults on this system. We undertake several simple analytical studies and make the following contributions. We show that:

- The time between uncorrectable memory faults over the lifetime of Cielo obey Benford's Law: the leading digit is more likely to be small (§3.2);
- The correctable faults from Cielo do *not* appear to obey Benford's law. We also outline a few suggestions as to why this is not true (§3.2); and
- Several common theoretical distributions used in HPC to model failures also appear to obey Benford's Law (§3.3).

To the best of our knowledge, this is the first work to demonstrate that memory faults from an large-scale HPC system obey a Benford distribution. It also provides critical analysis on the occurrence of memory failures on extreme-scale systems. Specifically, our analysis could be used to improve existing failure prediction mechanisms or to make models of memory failures more realistic.

2 Background

2.1 System Description

Cielo was a leadership-class HPC system located in Los Alamos, New Mexico. It was a Cray XE6 system running Linux that was operated from March 2011 to May 2016. At the time of its decommissioning, it was comprised of approximately 8,500 compute nodes. Each compute node contained 32 GB of DRAM and two processor sockets, each occupied by an AMD Opteron^{TM} 8-core processor. Cielo consisted of 96 *racks* of compute nodes arranged in 6 rows. Each rack contained 96 compute nodes arranged in a three-level hierarchy. Each rack was composed of three *chassis*. Each chassis was composed of eight *slots*. Each slot hosted four compute nodes.

2.2 Terminology: faults and errors

Throughout this paper, we distinguish between faults and errors, *cf.* [2]. A **fault** is the underlying cause of an error (e.g., stuck-at bits or high-energy particle strikes). An **error** is incorrect system state due to an active fault. Errors are *detected* and possibly *corrected* by higher-level mechanisms such as parity or error correcting codes (ECC). They may also be *uncorrected* or, in the worst case, *undetected*.

2.3 Terminology: Transient Vs. Permanent Faults

Hardware faults can be classified as *transient*, *intermittent*, or *hard* [3] [6] [7]. *Transient faults*, which cause incorrect data to be read from a memory location until the location is overwritten with correct data. These faults occur randomly and are not indicative of device damage [3]. Particle-induced upsets ("soft errors"), which have been extensively studied in the literature [3][27], are one type of transient fault. Distinguishing a hard fault from an intermittent fault in a running system requires knowing the exact memory access pattern to determine whether a memory location returns the wrong data on every access. In practice, this is impossible in a large-scale field study such as ours. Therefore, we group intermittent and hard faults together in a category of *permanent* faults.

2.4 Memory Failure Logs

All of the DRAM on Cielo is protected by chipkill-correct ECC. When the memory controller detects a memory error, it is designed to use ECC to correct the error. If it is able to correct the error, the error is recorded as a *correctable error* (CE). If it is unable to correct the error, the error is recorded as a *detected, uncorrectable error* (DUE). Correctable errors are recorded in registers provided by the x86 Machine Check Architecture (MCA) [1]. The contents of these registers are polled periodically and written to the console log. Uncorrectable errors are recorded in an event log after the node is rebooted. For both correctable and uncorrectable errors, detailed information about each error is recorded. This information includes the physical address where the error occurred and ECC syndrome data that describes the cause of the error. Decoding the recorded information about each error allows us to identify the physical location of each logged error. We examined the memory error logs collected on Cielo from May 2011 to May 2016. Additional details can be found elsewhere [16, 23].

2.5 Benford's Law

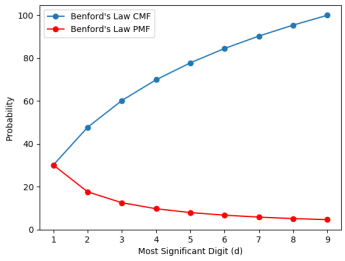


Fig. 1. Probability mass function for Benford distribution and cumulative mass function (CMF)

Benford's law, also called the Newcomb–Benford law, the law of anomalous numbers, or the first-digit law, is an observation about the frequency distribution of leading digits in many real-life sets of numerical data. The law states that in many naturally occurring collections of numbers, the leading digit is likely to be small. In sets that obey the law, the number 1 appears as the leading significant digit about 30% of the time, while 9 appears as the leading significant digit less than 5% of the time. The law is named after physicist Frank Benford, who proposed the law in 1938 [4], although it had been previously observed by Simon Newcomb in 1881 [20]. Benford's Law has been shown to apply to a wide variety of data sets, including electricity bills, street addresses, stock prices, house prices, death rates, lengths of rivers, and physical and mathematical constants.

Mathematically, the probability distribution of the leading digit d ($d \in \{1, \cdots, 9\}$ is:

$$P(d) = \log_{10}(d+1) - \log_{10}(d) = \log_{10}\left(\frac{d+1}{d}\right) \tag{1}$$

Figure 1 shows both the probability distribution function (PDF) and the cumulative distribution function (CDF) for a theoretical Benford distribution.

3 Experimental Results

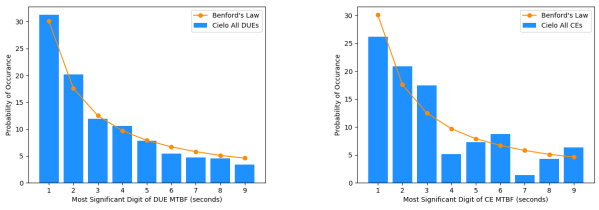
3.1 Methodology

In the following sections we calculate the probability mass function of the leading digit and compare with a theoretical Benford distribution. For this calculation we use the time between memory faults in seconds. If the first digit of the time between faults begins with a zero, we use the first non-zero digit in the calculation.

The choice of seconds is arbitrary as the properties of this distribution is independent of the representations (*i.e.*, if an observation obeys Benford's Law it does not matter how that metric is represented). More formally, Benford's Law has been shown to be sum-invariant, inverse-invariant, and addition and subtraction invariant [13, 5].

3.2 Cielo System Lifetime Data Benford Analysis

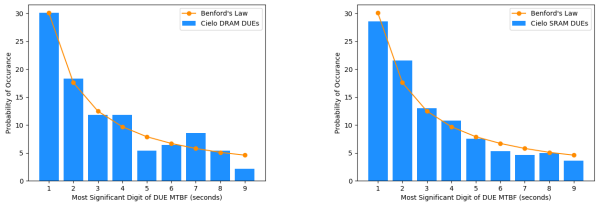
Figure 2 shows the empirical distribution of the first digit of the time between faults in DRAM and SRAM over the lifetime of Cielo, measured in seconds. Figure 2a shows the data for uncorrectable memory faults. Figure 2b shows the data for correctable memory faults. From this figure, we make a few important observations. First, the intervals between uncorrectable memory faults follows a Benford distribution: memory fault intervals are much more likely to have a small first digit. However, Figure 2b shows that while the interval between correctable memory faults are more likely to have a small first digit, they do *not* follow a Benford distribution as closely as the uncorrectable memory faults do.



(a) All Uncorrectable Faults on Cielo (b) All Correctable Faults on Cielo

Fig. 2. Benford distribution of fault time for all correctable faults and uncorrectable faults over the entire lifetime of Cielo

This difference may be explained by the mechanics of how errors are logged on the system. As described in Section 2.4 correctable memory faults on Cielo were logged in a ring buffer. Therefore, it is possible that some of these errors were lost when the ring buffer overflowed during computation. This is due, at least in part, to the fact that a correctable memory fault can produce a very large number of correctable errors, depending on the system's memory access patterns. If errors are lost, the calculation of the fault time may be affected. In contrast, it is less likely that uncorrectable memory faults are lost because the affected node halts and the single fault is recorded.



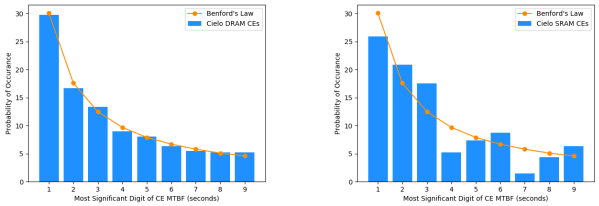
(a) DRAM Uncorrectable Faults on Cielo (b) SRAM Uncorrectable Faults on Cielo

Fig. 3. Benford distribution of uncorrectable fault time for Static Random Access Memory (SRAM) and Dynamic Random Access Memory (DRAM) on Cielo

To understand how these data are affected by memory technology, Figure 3 shows the leading digits of our uncorrectable memory fault data divided into two groups: memory faults in Static Random Access Memory (SRAM) (Figure 3b); and memory faults in Dynamic Random Access Memory (DRAM) (Figure 3a). Investigating these differences are important to developing a complete understanding of how memory faults occur because these two memory technologies

use different protection mechanisms on Cielo; Chipkill [9] is used to protect DRAM, and memory parity is used to protect SRAM.

From the data in these figures. we make several observations. First, the SRAM uncorrectable fault times in Figure 3b appear to follow a Benford distribution. The likely reason for this is due to total number of faults in each of these two scenarios. Because some of the logs we analyzed contain confidential information, we cannot comment on the total number of DRAM or SRAM faults, but over its lifetime, Cielo experienced more SRAM errors in comparison to DRAM. This is related to the fact that the SRAM structures are typically protected only by parity. Recent AMD processors provide much stronger SRAM protection. Finally, we observe that although the Benford distribution does not appear to be a good match for the intervals between uncorrectable DRAM faults, they do exhibit a similar trend: leading digits are still likely to be small.



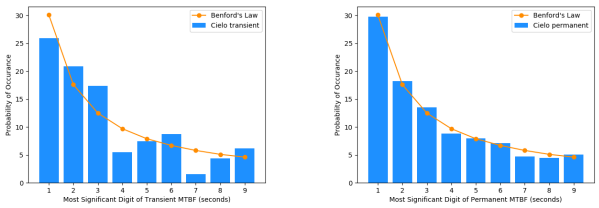
(a) DRAM Correctable DRAM Faults on Cielo

(b) SRAM Correctable DRAM Faults on Cielo

Fig. 4. Benford distribution of correctable fault times for Static Random Access Memory (SRAM) and Dynamic Random Access Memory (DRAM) on Cielo

In Figure 4, we examine the same data for correctable memory faults. Interestingly, we observe a trend that is the opposite of what we observed with uncorrectable memory faults. Specifically, the Benford distribution is a good match for the time between DRAM correctable faults while the match between the Benford distribution and the time between correctable SRAM faults is not particularly good. In this case, these differences in SRAM cannot be attributed to the size of the data sample (*i.e.*, the total number of correctable memory faults in our dataset). Correctable faults are much more common than uncorrectable faults. As a result, we do not believe that these results can be attributed to the size of the sample. We are currently investigating the source of this phenomenon. As with uncorrectable memory faults, it might be related to the differences of logging and reporting the correctable errors. However, further study is needed.

Finally, Figure 5 shows the distribution of failure interarrival times for both permanent and transient faults. For the data in these figures, we only distinguish between faults based on whether they are transient or permanent. All other distinctions are ignored; each dataset includes SRAM and DRAM faults, and correctable and uncorrectable memory faults). From this figure we observe that



(a) Transient Faults (b) Permanent Faults

Fig. 5. Benford distribution of the interarrival times for permanent and transient faults on Cielo.

the permanent errors more closely follow a Benford distribution. This result may not be surprising given the fact that the majority of permanent faults are uncorrectable and transient faults are more likely to be correctable. However, it suggests that further analysis is needed to understand if the processes behind these faults obey a Benford distribution. Although the Benford distribution is not a particularly fit for the intervals between transient memory faults, these intervals do exhibit the same general trend: small leading digits are more common than large leading digits.

3.3 Theoretical Distributions

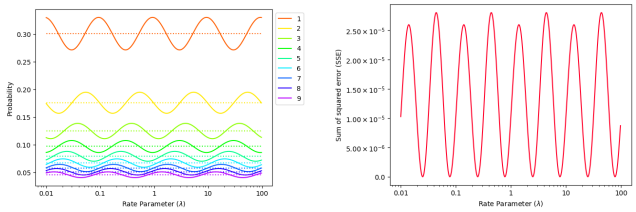
In the previous section, we observed that fault interarrival time for Cielo appeared to follow closely a Benford distribution. In addition to charactertizing and tabulating failures, fitting failures to known distributions is common in fault tolerance. In this section, we examine the relationship between Benford's Law and three probability distributions that are commonly used to model failures on HPC systems: exponential, Weibull, and gamma.

Mathematically, the probability mass function of the leading digit d ($d \in \{1, \cdots, 9\}$ for a theoretical probability distribution is:

$$P(d) = \sum_{k=-\infty}^{\infty} \Big(F((d+1) \cdot 10^k) - F(d \cdot 10^k) \Big)$$

where $F(x)$ is a cumulative density function (CDF).

Figure 6a shows the probability of the leading digit of a random variable drawn from exponential distributions. The solid lines represent the probabilities based on the theoretical distribution. The dashed lines represent the probability predicted by Benford's Law. Figures 7a and 7c show the same data for two different groups of Weibull distributions, corresponding to two different values of the shape parameter (0.25 and 0.75). Figures 8a and 8c show the same data for two different groups of gamma distributions, corresponding to two different values of the shape parameter (0.25 and 0.75).



(a) **Per-digit probability**. (b) **Sum of Squared Errors (SSE)**

Fig. 6. Exponential Distribution. Comparison of the probability leading digits from data drawn from exponential distributions to the results predicted by Benford's Law. In subfigure (a), solid lines represent values for the theoretical exponential distributions. Dashed lines represent the values predicted by Benford's Law.

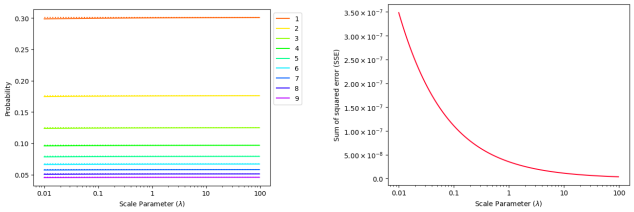
Figure 6b shows the the sum of squared errors (SSE) of the leading digit probabilities based on theoretical exponential distributions relative to the probability predicted by Benford's Law. Figures 7b and 7d show the same data for two different groups of Weibull distributions, corresponding to two different values of the shape parameter (0.25 and 0.75). Figures 8b and 8d show the same data for two different groups of gamma distributions, corresponding to two different values of the shape parameter (0.25 and 0.75).

These figures show that the probability of leading digits for random variables drawn from these theoretical distributions closely match the values predicted by Benford's Law. Because these distributions have been shown to be a reasonable fit for memory errors on Cielo [16], these data help explain why Benford's Law accurately predicts the leading digits of these intervals on Cielo.

4 Related Work

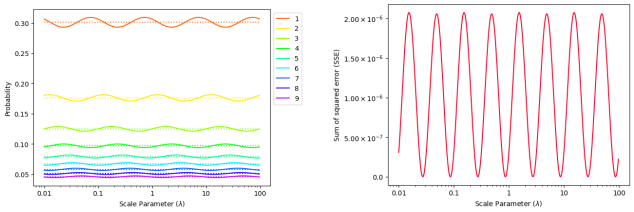
Failures characterization on large computer systems has been ongoing for over a decade. These studies have focused both on failures in HPC centers [21, 25, 24, 12, 26, 10, 23, 16, 11] and industry datacenters [22, 18, 17, 12]. These studies cover a wide diversity of systems of varying sizes and hardware/software configurations, yet many common failure trends are observed across all these systems.

Our work distinguishes itself from the existing studies in a number of important ways. First, to the best of our knowledge this is the first study to examine Benford's law and the interarrival times of failures for an HPC system. Second, this work is critical to both those modeling faults for HPC and those studying failures as it provides a simple methodology for verifying failure times. Finally, the results of this work may be of use to the those trying to mitigate or predict failures as this Benford property might be utilized to aid failure prediction.



(a) **Per-digit probability** (Weibull w/ shape (k) = 0.25)

(b) **Sum of Squared Errors (SSE)** (Weibull w/ shape (k) = 0.25)



(c) **Per-digit probability** (Weibull w/ shape (k) = 0.75)

(d) **Sum of Squared Errors (SSE)** (Weibull w/ shape (k) = 0.75)

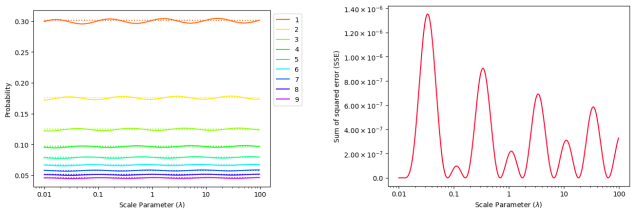
Fig. 7. Weibull Distribution. Probability of leading digits from data drawn from two groups of Weibull distributions (each with a different value of the shape parameter) to the results predicted by Benford's Law. In subfigures (a) and (c), solid lines represent values for the theoretical Weibull distributions. Dashed lines represent the values predicted by Benford's Law.

5 Conclusions

In this paper, we have provided a study of the time interval between memory faults for both correctable and uncorrectable errors on the Cielo supercomputer that was located at Los Alamos National Laboratory. Through this analysis, we show that the time between uncorrectable faults for this system obeys Benford's law – a law that states that the leading digits of some naturally occurring datasets is more likely to be small in value. We also show that correctable errors do not appear to follow this law, possibly due to the fact that the logging of correctable errors is done by polling mechanism and therefore many errors can be missed or logged with times that vary significantly from the actual fault time. Finally, we show that many common distributions used in literature to model failures also follow a Benford distribution.

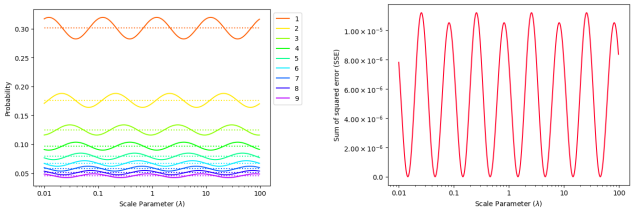
Acknowledgment

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly



(a) **Per-digit probability** (gamma with shape $(k) = 0.25$)

(b) **Sum of Squared Errors (SSE)** (gamma w/ shape $(k) = 0.25$)



(c) **Per-digit probability** (gamma w/ shape $(k) = 0.75$)

(d) **Sum of Squared Errors (SSE)** (gamma w/ shape $(k) = 0.75$)

Fig. 8. Gamma Distribution. Probability leading digits from data drawn from two groups of gamma distributions (each with a different value of the shape parameter) to the results predicted by Benford's Law. In subfigures (a) and (c), solid lines represent values for the theoretical gamma distributions. Dashed lines represent the values predicted by Benford's Law.

owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

References

1. AMD64 architecture programmer's manual volume 2: System programming, revision 3.23. `http://developer.amd.com/wordpress/media/2012/10/24593_APM_v21.pdf` (2013)
2. Avizienis, A., Laprie, J.C., Randell, B., Landwehr, C.: Basic concepts and taxonomy of dependable and secure computing. Dependable and Secure Computing, IEEE Transactions on **1**(1), 11–33 (2004). https://doi.org/10.1109/TDSC.2004.2
3. Baumann, R.: Radiation-induced soft errors in advanced semiconductor technologies. IEEE Transactions on Device and Materials Reliability **5**(3), 305–316 (Sept 2005). https://doi.org/10.1109/TDMR.2005.853449
4. Benford, F.: The law of anomalous numbers. Proc. Am. Philos. Soc. **78**(4), 551–572 (March 1938)
5. Berger, A., Hill, T.P.: Benford's law strikes back: no simple explanation in sight for mathematical gem **33**(1), 85–91 (Mar 2011).

https://doi.org/https://doi.org/10.1007/s00283-010-9182-3, http://link.springer.com/article/10.1007/s00283-010-9182-3

6. Constantinescu, C.: Impact of deep submicron technology on dependability of VLSI circuits. In: Dependable Systems and Networks, 2002. DSN 2002. Proceedings. International Conference on. pp. 205–209 (2002). https://doi.org/10.1109/DSN.2002.1028901
7. Constantinescu, C.: Trends and challenges in VLSI circuit reliability. IEEE Micro **23**(4), 14–19 (2003). https://doi.org/http://dx.doi.org/10.1109/MM.2003.1225959
8. Daly, J.T.: A higher order estimate of the optimum checkpoint interval for restart dumps. Future Generation Computing Systems **22**(3), 303–312 (2006). https://doi.org/http://dx.doi.org/10.1016/j.future.2004.11.016
9. Dell, T.J.: A white paper on the benefits of chipkill-correct ECC for PC server main memory. IBM Microelectronics Division pp. 1–23 (1997)
10. Di Martino, C., Kalbarczyk, Z., Iyer, R.K., Baccanico, F., Fullop, J., Kramer, W.: Lessons learned from the analysis of system failures at petascale: The case of Blue Waters. In: International Conference on Dependable Systems and Networks (2014)
11. Gupta, S., Patel, T., Engelmann, C., Tiwari, D.: Failures in large scale systems: Long-term measurement, analysis, and implications. In: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis. pp. 44:1–44:12. SC '17, ACM, New York, NY, USA (2017). https://doi.org/10.1145/3126908.3126937, http://doi.acm.org/10.1145/3126908.3126937
12. Hwang, A.A., Stefanovici, I.A., Schroeder, B.: Cosmic rays don't strike twice: understanding the nature of DRAM errors and the implications for system design. In: Proceedings of the 17th international conference on Architectural Support for Programming Languages and Operating Systems. pp. 111–122. ASPLOS XVII, ACM, New York, NY, USA (2012). https://doi.org/10.1145/2150976.2150989, http://doi.acm.org/10.1145/2150976.2150989
13. Jamain, A.: Benford's Law. Master's thesis, Department of Mathematics, Imperial College of London and ENSIMAG, London, UK (2001), http://www.math.ualberta.ca/~aberger/benford_bibliography/jamain_thesis01.pdf, not found in Imperial College Library or COPAC catalogs on 16 February 2013. URL link is broken too.
14. Jauk, D., Yang, D., Schulz, M.: Predicting faults in high performance computing systems: An in-depth survey of the state-of-the-practice. In: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis. SC '19, Association for Computing Machinery, New York, NY, USA (2019). https://doi.org/10.1145/3295500.3356185, https://doi.org/10.1145/3295500.3356185
15. Kondo, D., Javadi, B., Iosup, A., Epema, D.: The failure trace archive: Enabling comparative analysis of failures in diverse distributed systems. In: Cluster, Cloud and Grid Computing (CCGrid), 2010 10th IEEE/ACM International Conference on. pp. 398–407. IEEE (2010)
16. Levy, S., Ferreira, K.B., DeBardeleben, N., Siddiqua, T., Sridharan, V., Baseman, E.: Lessons learned from memory errors observed over the lifetime of cielo. In: Proceedings of the International Conference for High Performance Computing, Networking, Storage, and Analysis. SC '18, IEEE Press (2018)
17. Li, X., Huang, M.C., Shen, K., Chu, L.: A realistic evaluation of memory hardware errors and software system susceptibility. In: Proceedings of the 2010

USENIX conference on USENIX annual technical conference. pp. 6–20. USENIXATC'10, USENIX Association, Berkeley, Calif., USA (2010), http://dl.acm.org/citation.cfm?id=1855840.1855846
18. Li, X., Shen, K., Huang, M.C., Chu, L.: A memory soft error measurement on production systems. In: 2007 USENIX Annual Technical Conference on Proceedings of the USENIX Annual Technical Conference. pp. 21:1–21:6. ATC'07, USENIX Association, Berkeley, Calif., USA (2007), http://dl.acm.org/citation.cfm?id=1364385.1364406
19. Liu, Y., Nassar, R., Leangsuksun, C., Naksinehaboon, N., Paun, M., Scott, S.L.: An optimal checkpoint/restart model for a large scale high performance computing system. In: Parallel and Distributed Processing, 2008. IPDPS 2008. IEEE International Symposium on. pp. 1–9. IEEE (2008)
20. Newcomb, S.: Note on the frequency of use of the different digits in natural numbers. American Journal of Mathematics **4**(1–4), 39–40 (1881), http://www.jstor.org/stable/2369148
21. Schroeder, B., Gibson, G.A.: A large-scale study of failures in high-performance computing systems. In: Proceedings of the International Conference on Dependable Systems and Networks. pp. 249–258. DSN '06, IEEE Computer Society, Washington, DC, USA (2006). https://doi.org/10.1109/DSN.2006.5, http://dx.doi.org/10.1109/DSN.2006.5
22. Schroeder, B., Pinheiro, E., Weber, W.D.: DRAM errors in the wild: a large-scale field study. Commun. ACM **54**(2), 100–107 (Feb 2009). https://doi.org/10.1145/1897816.1897844, http://doi.acm.org/10.1145/1897816.1897844
23. Siddiqua, T., Sridharan, V., Raasch, S.E., DeBardeleben, N., Ferreira, K.B., Levy, S., Baseman, E., Guan, Q.: Lifetime memory reliability data from the field. In: 2017 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT). pp. 1–6 (Oct 2017). https://doi.org/10.1109/DFT.2017.8244428
24. Sridharan, V., DeBardeleben, N., Blanchard, S., Ferreira, K.B., Stearley, J., Shalf, J., Gurumurthi, S.: Memory errors in modern systems: The good, the bad, and the ugly. In: Proceedings of the Twentieth International Conference on Architectural Support for Programming Languages and Operating Systems. pp. 297–310. ASPLOS '15, ACM, New York, NY, USA (2015). https://doi.org/10.1145/2694344.2694348, http://doi.acm.org/10.1145/2694344.2694348
25. Sridharan, V., Liberty, D.: A study of DRAM failures in the field. In: Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis. pp. 76:1–76:11. SC '12, IEEE Computer Society Press, Los Alamitos, CA, USA (2012), http://dl.acm.org/citation.cfm?id=2388996.2389100
26. Sridharan, V., Stearley, J., DeBardeleben, N., Blanchard, S., Gurumurthi, S.: Feng shui of supercomputer memory: Positional effects in DRAM and SRAM faults. In: Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis. pp. 22:1–22:11. SC '13, ACM, New York, NY, USA (2013). https://doi.org/10.1145/2503210.2503257, http://doi.acm.org/10.1145/2503210.2503257
27. Ziegler, J., Lanford, W.: The effect of sea level cosmic rays on electronic devices. Journal of Applied Physics **52**(6), 4305–4312 (1981). https://doi.org/10.1063/1.329243