SAND2022-6083C

Understanding Memory Failures on a Petascale Arm System

Kurt B. Ferreira, Scott Levy, Joshua Hemmert & Kevin Pedretti
{kbferre,sllevy,jhemmer,ktpedre}@sandia.gov
Center for Computing Research
Sandia National Laboratories
Albuquerque, New Mexico, USA

ABSTRACT

New and novel HPC platforms provide interesting challenges and opportunities. Analysis of these systems can provide a better understanding of both the specific platform being studied as well as large-scale systems in general. Arm is one such architecture that has been explored in HPC for several years, however little is still known about its viability for supporting large-scale production workloads in terms of system reliability. The Astra system at Sandia National Laboratories was the first public peta-FLOPS Arm-based system on the Top500 and has been successfully running production HPC applications for a couple of years. In this paper, we analyze memory failure data collected from Astra while the system was in production running unclassified applications. This analysis revealed several interesting contributions related to both the Arm platform and to HPC systems in general. First, we outline the number of components replaced due to reliability issues in standing-up this first-of-its-kind, large-scale HPC system. We show the distribution differences between correctable DRAM faults and errors on Astra, showing that, not properly accounting for faults can lead to erroneous conclusions. Additionally, we characterize DRAM faults on the system and show contrary to existing work that memory faults are uniformly distributed across CPU socket, DRAM column, bank and rack region, but are *not* uniform across node, DIMM rank, DIMM slot on the motherboard, and system rack: some racks, ranks and DIMM slots experience more faults than others. Similarly, we show the impact of temperature and power on DRAM correctable errors. Finally, we make a detailed comparison of results presented here with the positional affects found in several previous large-scale reliability studies. The results of this analysis provide valuable guidance to organizations standing-up first-in-class platforms in HPC, organizations using Arm in HPC, and the entire large-scale HPC community in general.

CCS CONCEPTS

• **Hardware → Failure recovery, maintenance and self-repair**; System-level fault tolerance; • **Computer systems organization → Reliability**.

KEYWORDS

Arm, DRAM Reliability, Memory Failures, Temperature Correlation, Hardware Infant Mortality

HPDC '22, June 27-July 1, 2022, Minneapolis, MN, USA
2022. ACM ISBN 978-1-4503-9199-3/22/06.
https://doi.org/10.1145/3502181.3531465

ACM Reference Format:
Kurt B. Ferreira, Scott Levy, Joshua Hemmert & Kevin Pedretti. 2022. Understanding Memory Failures on a Petascale Arm System. In *Proceedings of the 31st International Symposium on High-Performance Parallel and Distributed Computing (HPDC '22), June 27-July 1, 2022, Minneapolis, MN, USA.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3502181.3531465

1 INTRODUCTION

In the continued march of High-Performance Computing (HPC) toward ever-increasing performance in pursuit of Exascale, many novel node architectures and technologies are being explored [26, 35]. These new and novel platforms provide interesting challenges and important opportunities due to their diversity of features. Analysis of these existing production systems can provide critical understanding of both the specific platform being studied, as well as large-scale HPC systems more generally. Arm is one such new platform currently being investigated for its ability to support HPC workloads.

The Astra system at Sandia National Laboratories was the first public peta-FLOPS, Arm-based system on the Top500. At 2592 nodes, Astra is sufficiently large to enable us to develop an understanding of the potential risks of deploying and utilizing an exascale Arm-based system. The scale of this system is believed to adequately balance cost with the ability to mitigate risk by proving the viability of Department of Energy (DOE) National Nuclear Security Administration (NNSA) workloads on these systems.

While Astra has demonstrated the viability of the platform and the Arm processor to support production workloads [24], little is known about its ability to provide the system reliability necessary to support large-scale production workloads. This is particularly relevant given the large number of memory channels found on Astra. In this paper, we analyze initial failure and environmental data from Astra that was collected while the system was in production running unclassified applications. The analysis in this paper focuses on memory reliability because this is the primary source of on-node hardware failure on this and previous large-scale HPC systems [13–15, 29, 31, 33, 34]. The results of our analysis provide valuable guidance to organizations standing-up first-in-class platforms in HPC, organizations using Arm in HPC, and the entire large-scale HPC community in general including the following contributions:

- To the best of our knowledge, the first memory failure study for a large-scale HPC Arm system. This is key to evaluating the performance of Arm as it is becoming more popular in HPC.
- A detailed tally of the hardware components replaced in the early hardware stabilization periods (the so-called infant

mortality period) (§3.1), key in understanding the cost of standing up a typical first-in-its-class, HPC system.
- A detailed analysis of the distribution differences between correctable DRAM faults and errors on Astra, showing that not properly accounting for faults can lead to erroneous conclusions (§3.2).
- A detailed analysis that shows, contrary to previous studies, memory faults are fairly uniform across CPU socket, DRAM column, DRAM bank, and region within a rack but are less uniform across node, DRAM slot, and cabinet within a system (§3.2 and §3.4). This result is important to designing effective failure mitigation methods.
- The impact of temperature and power on DRAM correctable errors. Contrary to previous work, we show that there is not a strong correlation between higher temperatures and correctable memory error rate (§3.3).
- A detailed comparison of results presented here with the positional effects found in several previous large-scale reliability studies (§3.4), helping place this work in context to other analyses.
- The public release of the failure and environmental data presented here for verification and use in the research community available at [7].

Similar to existing works[13, 31, 33, 34], we utilize a well-established methodology for our reliability study. First, we extract relevant reliability information from the various system logs. Then, we process these extracted logs to reach the conclusions described in this paper. This work differs from previous work in several important ways. First, this is the first public study of a large-scale Arm-based HPC system. Second, along with the reliability data, we analyze detailed environmental data collected on Astra. Finally, the reliability and environmental data discussed in this work will be made publicly available at [7], following publication.

While we believe this work makes several significant contributions, a few limitations exist. First, the results in this paper are for a specific Arm-based HPC platform running a specific set of workloads that are specific to Sandia National Laboratories. While we believe this is an appropriate proxy for other Arm-based HPC systems, results from systems with significantly different usage patterns and environmental conditions may vary considerably. Additionally, the reliability of low-level system components can vary significantly by manufacturer [34] and date of manufacture. As a result, extreme care is required when trying to use our to data to predict the behavior and performance of similar systems.

2 BACKGROUND

In this section, we provide background information on our methodology and on the system we used to collect the data that we analyze in this paper.

2.1 Terminology

In this work, we distinguish between a fault and an error as follows [1]. A *fault* is the underlying cause of an error, such as a stuck-at bit. Faults can be active (leading to errors), or dormant (not causing errors). An *error* is incorrect state resulting from an active fault. Errors may be *detected* and possibly *corrected*, called a correctable error (CE)) by mechanisms such as parity or error correcting codes (ECC). They may also go uncorrected, called a detected uncorrectable error (DUE). Errors can also be completely undetected (i.e., silent), but these are beyond the scope of this work.

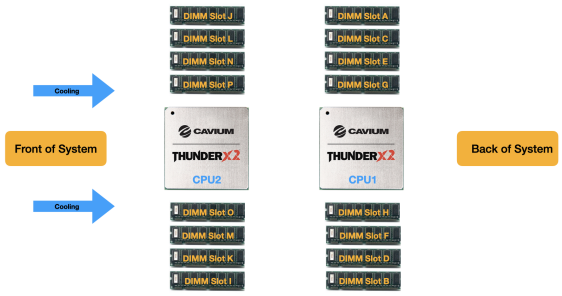


Figure 1: Physical layout of DIMM and CPU slots on Astra. Unlike some older supercomputers (e.g., Cielo), cooling in Astra flows from front to back rather than bottom to top.

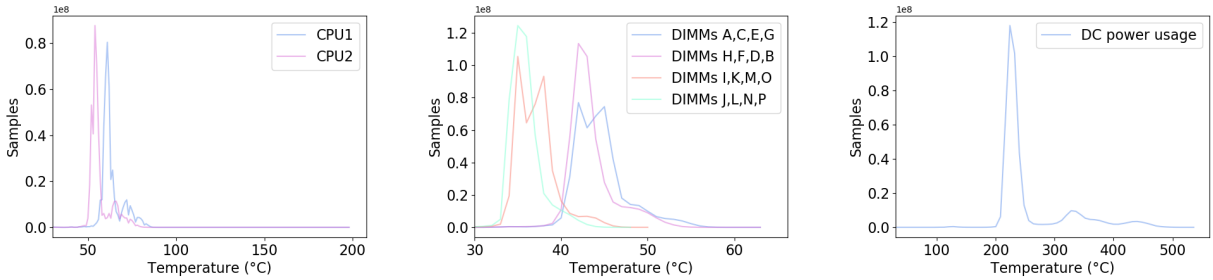
Similar to prior studies, we identify several unique DRAM fault modes: single-bit, in which all errors map to a single bit; single-word, in which all errors map to a single word; single-column, in which all errors map to a single column; single-row, in which all errors map to a single row of DRAM; and single-bank, in which all errors map to a single DRAM bank.

2.2 The Astra Platform Details

Astra [24] consists of 2,592 dual-socket compute nodes totaling 145,152 cores with an aggregate theoretical peak compute performance of 2.3 PF/s. Astra consists of 36 racks containing 72 compute nodes each. Each rack on Astra contains 18 chassis, with each chassis containing 4 nodes. Astra was designed to be well-balanced and large enough to attract a broad set of users with diverse applications to the platform. An overarching goal of the project was to demonstrate the viability of the Arm architecture for supporting NNSA large-scale HPC modeling and simulation workloads.

Each Astra compute node employs two sockets, each with a 28-core Marvell CN9975-2000 ThunderX2 processor [19] running at 2.0 GHz. One of the key features of the node architecture is the inclusion of eight memory channels per socket, versus the typical six offered by comparable general-purpose processors available at the time of its procurement. By utilizing 8 GB DDR4-2666 dual-rank registered DIMMs, one DIMM per memory channel, the resulting aggregate memory capacity of Astra reaches 332 TB with an aggregate memory bandwidth of 885 TB/s. Unlike many HPC platforms of its size, Astra does not utilize Chipkill [4] to protect the contents of its DRAM; it uses the cheaper and less power-hungry single-error-correction, double-error-detection (SEC-DED) ECC.

Each compute node of Astra includes six temperature sensors: one CPU temperature sensor and two DIMM temperature sensors per socket. The DIMM sensors are each positioned to measure the temperature for a group of 4 DIMM slots: DIMM slots A, C, E and



(a) CPU Temperature Distributions **(b) DIMM Temperature Distributions** **(c) Node DC Power Usage Distribution**

Figure 2: Histogram of sensor values from May 20 to September 19, 2019.

G share one temperature sensor; DIMM slots I, K, M and O share a sensor; DIMM slots H, F, D and B share a sensor; and DIMM slots J, L, N and P share a sensor. Each compute node also has a sensor that measures the DC power consumed by the node. Data from these sensors are collected once per minute and stored in a back-end database. The physical layout of the DIMM and CPU slots relative to how the system is cooled is shown in Figure 1.

The distribution of the data collected from these sensors is shown in Figure 2. For all of our sensor data, there are instances where the sensors were either not functioning or were not properly read. For obvious reasons, we exclude these data points from our analysis. Additionally, there were several instances where the DC power sensors recorded values that were clearly identified as invalid. We also exclude these values from our analysis. In all of our datasets, the number of excluded sensor samples was significantly less than 1% of the total.

2.3 Error Logging

Correctable errors are logged internally, with space for a limited number of errors. Once logging space is full, further CEs may be dropped. This logging space is read periodically by the operating system via a polling mechanism that runs every few seconds. Once read, the details of the CE are written to the syslog. Uncorrectable errors are recorded via a machine check and logged to the syslog or serial console depending on the severity. This typically means that uncorrectable errors are seldom lost, unlike correctable errors.

Unless otherwise noted, our failure analysis spans an interval of time from January 20, 2019 to September 14, 2019 when the system was moved to a closed network. During this time, Astra was undergoing a production stabilization period where users were encouraged to stress the machine to shake out hardware and software issues.

2.4 Open Source Data for Astra

As stated previously, both the memory error and associated environmental data used in this work will be available upon publication. Specifically, we will provide text files containing both the memory failure telemetry information extracted from the system logs and the environmental sensor data extracted from the baseboard management controller (BMC) log files. The failure data includes a timestamp, node ID, socket, type of failure, DIMM slot, row, rank, bank, bit position, physical address and vendor-specific syndrome data. For environmental data, we include per-node power draw and temperature readings for 6 sensors located on each node, *see* Section 2.2 for details on the available temperature sensors. Data was collected from each sensor once per minute and a timestamp was included for each reading. The total volume of the data analyzed in this paper is approximately 8 GiB. These data will be made available at [7].

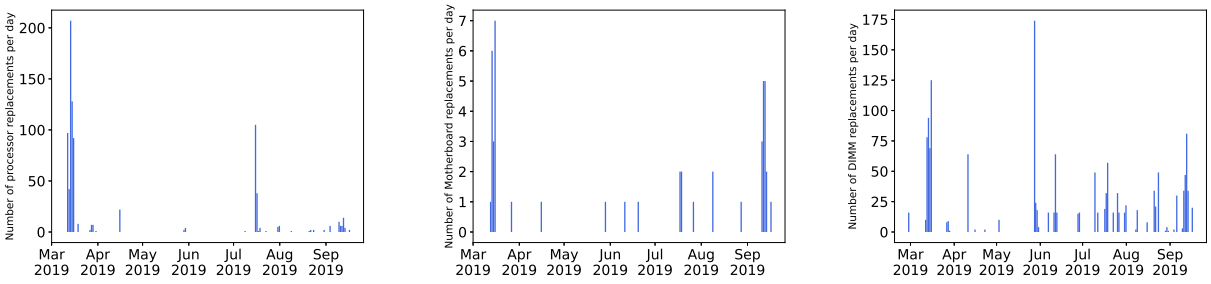
Table 1: Astra component replacements from Feb 17, 2019 to Sep 17, 2019.

Component	Number Replaced	Percent of Total
Processors	836	16.1% of 5184
Motherboards	46	1.8% of 2592
DIMMS	1515	3.7% of 41472

3 RESULTS

3.1 Hardware Replacement due to Reliability

We first look at the hardware replacements on Astra to get an idea of initial reliability and hardware infant mortality. We believe that this data is not indicative of low quality parts or poor quality control, but is typical of a first-in-class architecture like Astra constructed from current technology parts [24]. Our experiences suggests these one-of-a-kind designs are largely field-tested at scale where additional engineering issues arise. Table 1 and Figure 3 show the major component hardware replacement numbers from Feb 17, 2019 to Sep 17, 2019 when system stabilization was underway on the system and before the system was relocated to a closed network. Component replacements were detected by analyzing the site's daily inventory scan logs. These are components that were deemed defective and impacting system performance/reliability. The possibly surprising result is the volume of processors in need of replacement. While we have little external data to compare to as this type of data is rarely publicly released, the belief in the field is likely the DIMMs are more frequently replaced than processors. For Astra, the number of processor replacements was elevated due to a memory controller speed upgrade that was performed in the field.



(a) Processor **(b) Motherboard** **(c) DRAM DIMM**

Figure 3: Distribution of hardware replacements by day. For each of these components, a significant number of replacements were performed during the initial bring-up period (i.e., infant mortality). Subsequent spikes in replacements occurred as component-specific issues were identified.

Not all of the processors were able to support the increased speed and thus required replacement. We believe the public release of this aggregate data is important to system designers and data-centers to get an idea of the total cost involved in standing up new and novel systems.

Figures 3a to 3c show the daily count of hardware replacements from our inventory tracking system for processors, motherboard, and DIMMs, respectively. For the motherboard and processor replacement data, we note two periods of heavy replacement that are not coincident in time. The first for both was the beginning of our tracking period which coincides with the commencement of the stabilization period and can be viewed as the initial hardware infant mortality. The second uptick occurred after several months of heavy use in both cases. For processors, the second uptick in replacements occurred due to the memory controller speed upgrades outlined previously. For motherboards, this second uptick in replacements occurred after several months of sustained use.

For the DIMM replacement data in Figure 3c, trends are not so easily described. First, we do see an increased infant mortality replacement rate with the first month of this testing period. Also, the daily replacement rates are quite high in the middle of this testing interval, likely due to several cooling issues that were addressed during this period. Lastly, we see a constant and consistent replacement trend in the later section of our testing period which we believe is due to normal aging of some memory parts under heavy use. For all figures, the replacements that occurred at the end of our testing period correspond to a time when vendor representatives were on-site to address hardware issues and were done in preparation of the system being moved to a closed network.

3.2 Correctable DRAM Faults and Errors

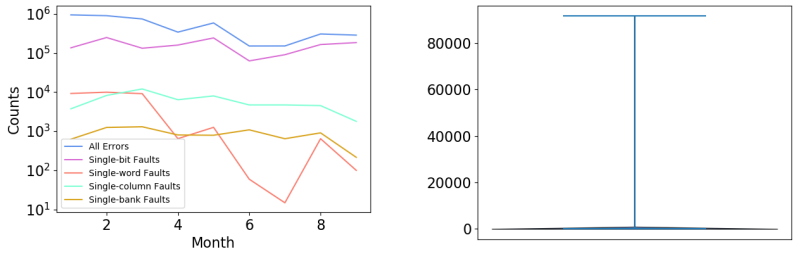
In this section we examine correctable DRAM faults and errors (*see* Section 2.1 for details) found on Astra. First we examine the total number of faults and errors on the system in this interval in time.

Astra DRAM Fault Modes. Figure 4a shows a breakdown of DRAM fault modes experienced in the Astra system. Similar to other studies in this area, we identify several unique DRAM fault modes: single-bit, in which all errors map to a single bit in the DRAM device; single-word, in which all errors map to a single word in the device; single-column, in which all errors map to a single column; and single-bank, in which all errors map to a single bank. Other studies also investigated single-row, multiple rank and multiple bank faults, but analysis for these rare errors is not possible on our system. For single-row, in this interval of time the system does not provide proper row information in the correctable error record passed to the `syslog`, so this analysis was not possible. For multiple-rank and multiple-bank errors, in our SEC-DED protected memory these errors would manifest as uncorrectable memory errors because of the number of corrupted bits.

Overall from this figure we see the system experienced over $4,369,731$ total correctable DRAM errors, or around six per node per day, on average. Of those errors, $1,412,738$ of them were single-bit faults, $31,055$ were single word faults, $54,126$ were single-column fault, and $7,658$ were single-bank faults. One other property to note is that in this interval of time when the Astra system was in a production environment, the number of faults show a slightly downward trend as time progresses. This demonstrates that good system administrative practices and advanced system software features, like page retirement [36], are effective at helping to maintain system reliability.

Figure 4b is a violin plot that gives an idea on the density of errors per fault for this period of time. From the figure we make two observations: 1) The vast majority of the correctable faults resulted in only one error, and 2) The maximum number of errors for a particular fault resulted in over $91,000$ errors.

The results of these two plots are significant and important to the field for a number of reasons. First, they demonstrate the vast majority of faults result in very few errors on current systems, this is in contrast to several previous works. This is significant as correctable errors, while still allowing the application to make further progress, can have significant performance implications [18, 24]. Second, the majority of the DRAM fault modes experienced on Astra have a small memory footprint. Mitigation methods like page-retirement [36] can easily map out small-footprint faults like single-bit and single-word faults without significant penalty to available system memory. However, single-bank errors can require



(a) Errors and Fault Modes **(b) Errors per Fault**

Figure 4: Errors and faults separated by mode and a Violin plot of the overall errors per fault on Astra. The maximum number of errors per fault is just over $91,000$, while the vast majority of errors (and the median) are one. This is due to memory reliability, effective mitigation mechanisms built into the operating system (*e.g.* page-retirement) and good system maintenance practices.

significant portions of memory address space to be mapped out, causing significant impacts.

Per-node Errors and Faults. Using the previous fault analysis, in the remainder of this section we analyze how correctable DRAM faults are distributed both on a macro-scale, throughout nodes on the entire system, and at a micro-scale, across individual DRAM cells.

Figure 5 shows the number of correctable faults and errors per node in this interval. From this figure we see that not all nodes experienced DRAM correctable errors, only 1013 nodes experienced at least one error. Similar to other work, we also see that the number of both faults and errors can vary dramatically from node to node. In order to better understand this correctable fault distribution, we will reorganize the data slightly differently.

In Figure 5a the x-axis is the number of observed faults on a particular node and the y-axis is the number of nodes in our dataset that experienced that number of correctable faults. From the figure we see that the vast majority of the nodes saw zero or one correctable faults. This also shows that the distribution of faults per node closely resembles a power law distribution [3].

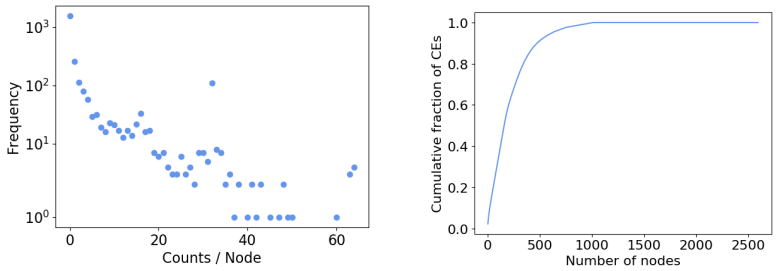
Figure 5b shows the empirical CDF of CEs by node. For each point (x, y) on the curve, the x nodes with most CEs represent a y fraction of the total CEs. These data show that a small fraction of nodes account for the overwhelming majority of the total number of CEs. For example, more than 60% of nodes experienced no CEs. The 8 nodes with the most CEs account for more than 50% of the overall total. The top 2% of nodes account for approximately 90% of the overall total number of CEs.

These results are significant to the study of failures for several reasons. First, the frequency and distribution shape is critical to modeling failures. Second, the relatively small number of faults per node suggest again that lightweight mechanisms for fault mitigation like page retirement and an exclude list for the small number of nodes experiencing large numbers of faults.

Per-socket, per-bank, and per-column Faults and Errors. To get a better understanding of correctable failures we will now look at how failures are distributed inside a node. Figures 6a to 6f show the distribution of correctable errors and faults across the sockets on a node, the memory banks on a DIMM, and the memory columns, respectively. If we just look at memory errors and not faults as many previous works do, we see we would get an inaccurate picture of how failures are distributed about a node. These data show that memory faults in these structures are fairly uniformly distributed and that variation can be explained by statistical noise. This result is consistent with the analysis by Sridharan et al. [34] of DRAM fault data collected on Cielo and Jaguar. In contrast, Hwang et al. [12] found that memory errors are more likely to occur on some columns and rows of memory than others. However, they only examine memory errors; they do not consider the associated memory faults. Our data shows a similar concentration of memory errors on some columns and banks but the phenomenon disappears if we examine memory faults instead.

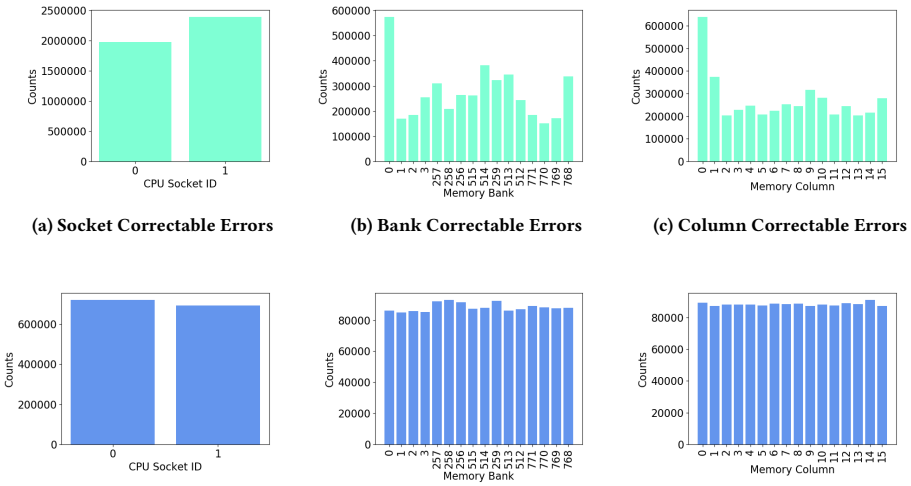
Per-rank and per-DIMM Slot Faults and Errors. While faults are uniformly distributed across several structures within a node, there are a few structures inside a node where the failure distribution is more irregular. Figures 7a to 7d show the number of errors and faults per rank on a node and per DIMM slot (*see* Section 2 for a description of slot layout). For Figures 7a and 7b, the relative occurrence of faults and errors is the same and rank zero seems to experience more faults (and errors). On Astra's DIMMs, a rank corresponds to all of the DRAM devices on one side of the DIMM. One possible reason one side of the DIMMs is experiencing higher error rates is differences in temperature (we will analyze the impact of temperature and failure in the next section). The physical layout of cooling on a node may make one side of the DIMM hotter than the other. For Figures 7c and 7d, we again see differences on the number of failures experienced per DIMM slot. Once again, this figure shows the importance of analyzing faults rather than errors. Additionally, we again see very different fault counts per slot, with DIMM slots J, E, I, P experiencing the greatest number of faults and DIMM slots A, K, L, M, and N experiencing the lowest number of faults. We theorize this difference may be due to potential temperature difference of the slots.

Per-bit Position Faults and Errors. Finally, we will look at the failure distributions of the bit position in a cache line that failed and



(a) Distribution of CE Counts per Node (b) CDF of the Distribution

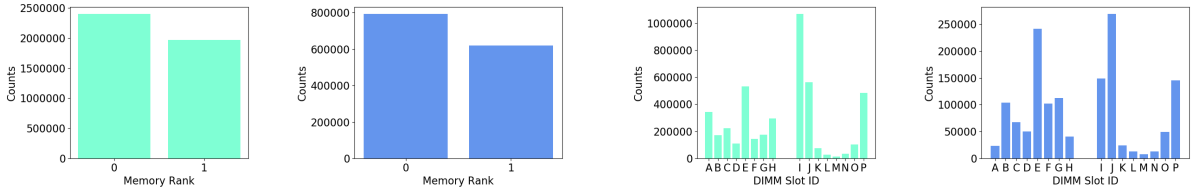
Figure 5: Number of correctable faults per node and empirical CDF of CEs by node. This distribution appears to obey a power law.



(a) Socket Correctable Errors (b) Bank Correctable Errors (c) Column Correctable Errors

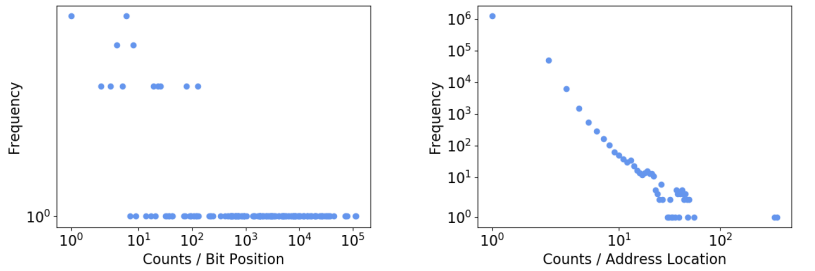
(d) Socket Correctable Faults (e) Bank Correctable Faults (f) Column Correctable Faults

Figure 6: Correctable errors and faults per CPU socket, bank, and column on Astra. Looking at errors, like many works in the field, would give an inaccurate picture of the distribution of failures inside a node.



(a) Correctable Errors per Rank (b) Correctable Faults per Rank (c) Correctable Errors per Slot (d) Correctable Faults per Slot

Figure 7: Correctable error and fault counts per memory rank and per DIMMM slot. Slots A–H are associated with socket 0, and I–P are associated with socket 1.



(a) Correctable Faults per Cacheline

(b) Correctable Faults per Physical Address

Figure 8: Number of correctable faults per cache line bit position and physical address. Both distributions appear to obey a power law.

the physical address of faults.[1] For brevity, we will only display the histograms of the fault counts per bit position and physical address. Figures 8a and 8b show the counts per bit position and physical address, respectively. We make two important observations. First, once again we see that for each bit position and physical address, the vast majority of locations see very few faults, similar to what we observed before. Also similar to before, these distributions appear to follow a power law, with the vast majority of locations being dominated by low fault counts. Again these observations are important for both modeling failures and mitigation method analysis.

3.3 Correlating Temperature and Correctable Error Rate

In the previous section, we showed that correctable errors and faults were distributed uniformly through some memory structures but not others. We speculated the reason for this may be due to the influence of temperature (higher temperature leading to more errors and faults). In this section, we examine the validity of that speculation.

For this analysis we will look at the interval of time from May 20, 2019 to September, 19, 2019. Note this interval of time is a subset of the data presented previously due to environmental data being missing from the previous interval. However, like before this is a period of time where the system was in production and being actively used.

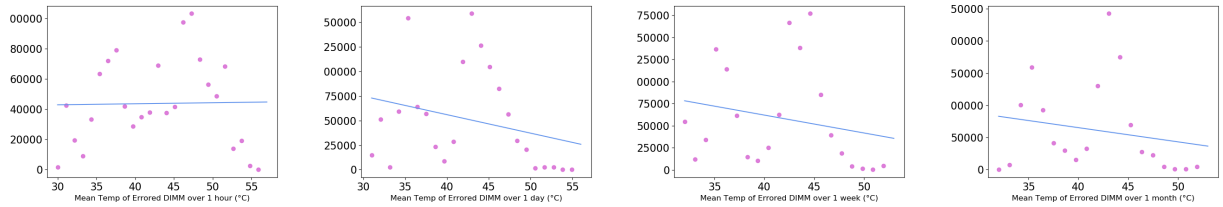
To analyze the impact of temperature on correctable errors we will do the following. For each correctable error, we used data from the DIMM sensor assigned to the DIMM on which the error occurred to calculate the mean temperature over the time interval immediately before the error is logged. We vary the duration of the interval over which the mean temperature is calculated from one hour to one month. The objective is to determine whether there is a temporal correlation between higher temperature and memory errors.

[1]The bit position portion of the CE record passed to the kernel seemed to encode additional data besides the actual failed bit position. While we could not decipher this additional encoded data, the encoding was consistent and therefore we believe does not impact our analysis

Figure 9 shows the correctable error counts for four different intervals prior to the error as a function of mean DIMM temperature. We computed the mean temperature of the affected DIMM over the interval immediately preceding the occurrence of the error. This figure shows results for intervals between one hour and one month. To examine the impact of temperature errors, we fit a line to the data points and observe the slope: a positive slope suggests higher temperatures prior to a correctable error lead to more frequent errors and a negative slope the opposite. Overall, the data in these figures show that increases in temperature is not strongly correlated with more frequent errors.

Schroeder et al. [28, 30] examined the relationship between temperature and correctable error rates (*see e.g.*, sections 4.1 and 4.2, and Figure 3 of Schroeder et al. [30]). They begin by plotting the monthly average temperature of each platform, in deciles versus the monthly correctable error rate within each decile. The temperature data available to them was collected every ten minutes from a single sensor on each motherboard in the system. Their data show that increasing the average temperature by $20°C$ is correlated with at least a doubling in the rate of correctable errors. Because of the correlation of temperature with utilization, they also divide the data into low temperature samples and high temperature samples and examine the relationship between CPU utilization and correctable error rate. Based on this data, they conclude that the increase correctable error rate can be explained by CPU utilization because the correctable error rate as a function of CPU utilization exhibits very similar behavior for the low temperature and high temperature datasets. The authors of these papers also consider the relationship of memory utilization on correctable error rate.

We performed the same analysis on our data to enable a direct comparison with the conclusions reached by Schroeder et al. Our dataset differs from the data in their papers in several important ways. First, we have much more detailed temperature information. Our data is collected once per minute from six total sensors per compute node: a CPU sensor and two DIMM sensors for each socket. Second, we do not have data that directly captures CPU utilization. As a proxy for CPU utilization, we used data collected from the compute node's DC power sensor which is also sampled once per minute. Third, we also do not have data that captures



(a) One Hour **(b) One Day** **(c) One Week** **(d) One Month**

Figure 9: Linear fit of CE error counts per average DIMM temperature for the interval immediately preceding the error (one hour, one day, one week, and one month). The trends in these data show that higher temperatures are not strongly correlated with more frequent errors.

memory utilization. Therefore, we have no way of examining the relationship between memory utilization and correctable error rate.

Figures 13a and 13b show the relationship between the temperature of each CPU and its DIMMs, respectively, and the correctable error rate. For each point (x, y) in these figures, x represents the maximum sample value within a decile and y represents the average monthly CE rate over the decile. As shown in Figure 1, cold air passes over the second socket (CPU2 and its DIMMs) before passing over the first socket (CPU1 and its DIMMs). Therefore, as shown in this figure, the temperature values collected from the first socket are generally higher than the temperature values collected from the second socket. These data suggest that the operating temperature on Astra was much more tightly controlled than on the systems examined by Schroeder et al. Schroeder et al. observed that the temperature difference between the first and ninth deciles was approximately $20°C$ for three of the systems and approximately $10°C$ for the fourth system. In contrast, our data shows the difference between the first and ninth deciles in our data is approximately $7°C$ for the CPUs and approximately $4°C$ for the DIMMs. However, because Schroeder et al. reports normalized temperatures, we cannot directly compare the absolute temperature recorded on Astra with the temperatures on the machines that they studied. In contrast to the conclusions reached by Schroeder et al., there is no discernible trend as the temperature increases; several of the lower temperature deciles have the highest observed correctable error rates. As a result, our data does not support Schroeder et al.'s conclusion that higher temperatures are correlated with more frequent correctable errors.

Figure 14 shows the relationship between utilization and correctable error rate. Each subfigure shows the utilization data divided in two and plotted independently. The utilization data in each subfigure is divided based on whether the associated temperature from the specified sensor is "high" (above the median temperature value for the specified sensor) or "low" (below the median temperature value for the specified sensor). Schroeder et al. used similar figures to isolate the impact of temperature from the impact of utilization. The basic idea is that by plotting the data this way they could compare the correctable error rate for samples that have the same utilization but different temperatures. As discussed above, our data does not include direct measurement of the CPU utilization on the nodes of Astra. However, we do have measurements of the input DC power for each node. We believe that these power measurements are a good proxy for CPU utilization: the more work that the CPU and other hardware components on the node do, the more power they require. As a result, we use input DC power to approximate utilization in the system. The data in Figure 14 shows that there is not a strong relationship between power use and correctable error rates: higher utilization does not correlate with more frequent correctable errors. These figures also show that there is a relationship between power use and temperature. This phenomenon is particularly evident in the data from the CPU sensors: the samples from the hot dataset have generally higher power usage (are shifted to the right relative to the data from the cold dataset). A similar, but less pronounced, effect is also present in the data collected from the DIMM sensors. Overall, this figure shows that, for the same power usage, hot samples frequently correspond to higher error rates than cold samples. However, this trend is far from universal and there are several cases where the reverse is true. Unlike the data relied on by Schroeder et al., the data in this figure do not support the conclusion that the impact of temperature on correctable errors is significantly smaller than the impact of CPU utilization.

Collectively, the data in Figures 13 and 14 show that, contrary to the data examined by Schroeder et al., there is not a strong relationship between temperature or power usage and correctable memory errors in our data. Although we cannot definitely determine why this difference exists, it is possible that it is due in part to the fact that the temperature on Astra was maintained within a much narrower range than on the systems analyzed by Schroeder et al. Moreover, we know that the absolute temperatures on Astra were maintained such that the temperature was never close to the devices' thermal limits. Unfortunately, we cannot make a direct comparison with the data analyzed by Schroeder et al. because they were unable to disclose any specific temperature data.[2] Similarly, El-Sayed et al. [6] examined uncorrectable DRAM errors and DRAM-related problems (i.e., node outages and hardware replacements) and concluded that the frequency of these events were not strongly correlated with temperature. Hsu et al. [11] reach a different conclusion; in their data, they find that compute node failures roughly double with

[2] Although we cannot make a direct comparison, the fact that there is a much wider temperature range in the data from Schroeder et al., more than $40°C$ between the first decile and the tenth decile (the maximum value) of their data compared to no more than $25°C$ between the first and tenth deciles in our data, suggests that their data contain higher overall temperature measurements.

each $10°C$ increase in temperature. However, because their analysis considers all causes of node unavailability, it is possible that this trend is due to other components in the system.

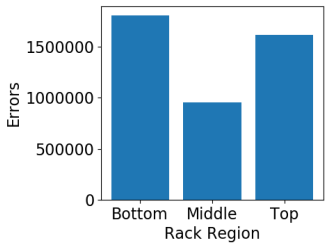
3.4 Positional Effects on Error Frequency

Sridharan et al. [34] examine how the physical location of compute nodes in Cielo and Jaguar may have affected the rate at which they experienced correctable SRAM errors. The compute nodes in each rack of Cielo were grouped into 3 chassis, arranged vertically. Their analysis revealed that the compute nodes in the chassis at the top of a rack experienced a higher rate of SRAM faults than compute nodes in the chassis at the bottom of the rate: approximately 20% more faults in the top chassis than in the bottom chassis. The authors hypothesize that one possible cause of this trend is the temperature differential within the rack. Cielo's cooling system was designed such that cool air entered the racks at the bottom through the floor of the machine room. Although the authors lacked detailed temperature logs to provide a detailed analysis of temperature trends within the rack, they did provide anecdotal evidence to support the claim that the chassis at the top of the rack generally were hotter than chassis at the bottom of the rack. They also speculated that there may be alternative explanations (e.g., cosmic rays) for the larger numbers of errors at the top of the racks. Similarly, Gupta et al. [10] found that node failures (from all causes) are more generally more likely for compute nodes in "cages"[3] that are closer to the top of the rack.

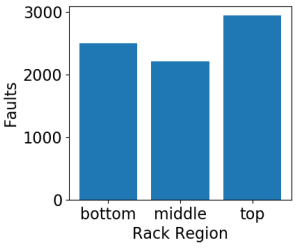
In contrast to Cielo and Jaguar (the two systems studied by Sridharan et al.), Astra's cooling system was designed so that cool air enters from the front of the rack and hot air is exhausted out of the back of the rack. The physical arrangement of Astra differs from those two systems (its racks contain 18 chassis stacked vertically rather than 3). Therefore, to facilitate a direct comparison with Sridharan et al.'s data, we divided each rack of Astra into three regions, each containing 6 chassis: top, middle, and bottom. We examined the temperature data collected on Astra averaged across the entire system for each of the three regions and for each of the six temperature sensors (data not included here due to space constraints). These data show that Astra did not exhibit the same temperature gradient within a rack as was observed in Cielo and Jaguar. In fact, these data show that the mean temperature is very consistent throughout the rack; there is no meaningful increase in temperature based on region (differences per region are significantly less than $1°C$). Because the mean temperature is so uniform throughout the rack, we can largely exclude it as a factor in any trends in memory errors within a rack.

Figures 10a and 10b shows the number of errors and faults, respectively, in each of the three regions within a rack. Note the difference again between the number and distribution of errors and faults. In the case of errors, the nodes at the bottom of the rack experienced the highest number of errors. The nodes at the top of the rack experience the second greatest number of memory errors. For faults that scenario is reversed, compute nodes near the top of the rack experience more frequent faults. However, the difference in the number of faults in each region is smaller than

[3]Based on the description of their system (Blue Waters), a cage is equivalent to a chassis in Cielo or Astra.



(a) Errors per region



(b) Faults per region

Figure 10: Errors and faults by rack position (region)

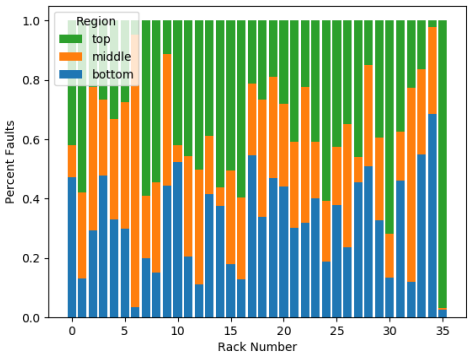
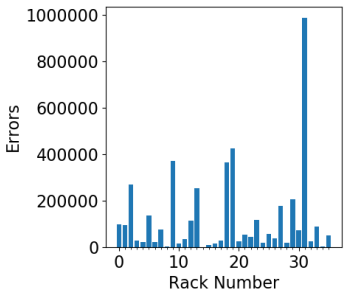
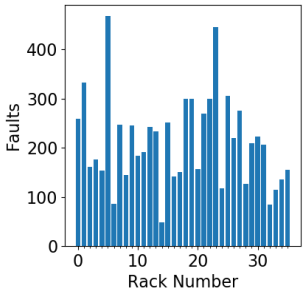


Figure 11: Percentage of faults per region by rack number

the difference in the number of errors in each region. Figure 11 provides a more detailed look at this data; it shows the fraction of faults in each rack that occurred in each region of the rack. These data show that faults are not significantly more likely to occur near the top of the rack than near the bottom of the rack. Broadly, we would expect the impact of cosmic rays to decrease more or less uniformly from the top of the rack to the bottom (i.e., the bottom of the rack is shielded by the top of the rack and by neighboring racks). As a result, although we cannot completely explain how a



(a) Errors per rack



(b) Faults per rack

Figure 12: Errors and faults by rack. These data show that: (i) a small number of faults may lead to a large number of errors; and (ii) the number of faults is not strongly correlated with rack position.

compute node's memory errors are affected by its position in the rack, our data does suggest that neither temperature nor cosmic rays adequately explain the relationship between which region of a rack a compute belongs to and the number of errors and faults that it experiences.

Sridharan et al. also observed that lower-numbered racks experienced more frequent errors than other racks in the system. Figures 12a and 12b shows the number of errors and faults, respectively experienced by each rack of Astra. There are several spikes in the error counts, e.g., Rack 31 experienced more than twice as many errors as any other rack. However, these spikes are not present in the fault data because large numbers of errors were caused by a relatively small number of faults. Figure 12b shows that there are no significant trends in the number of faults experienced by each rack. Similarly, El-Sayed et al. [6] found that rack position is not strongly correlated with uncorrectable memory errors.

Sridharan et al. also speculated that trends in memory errors per rack could be caused by temperature variation before concluding that the trends in their data could be explained by differences in reliability of DRAM devices from different vendors. [10] et al. similarly speculate that environmental factors, including temperature, may account for the non-uniform distribution of failures in their data. To understand the extent to which rack-to-rack temperature variation existed in Astra, we computed the mean temperature within each rack for each of the six available compute node temperature sensors. We found that the mean temperature for each sensor varied very little (less than approximately $4.2°C$) across the racks of the system. The consistent temperature across the racks of Astra may help explain why the number of memory faults are comparatively evenly distributed, *cf.* Figure 12b.

3.5 Uncorrectable Errors

Uncorrectable memory errors occur when the memory controller is able to use the ECC to determine that an error occurred but it is unable to recover the correct value. On Astra, uncorrectable memory errors are recorded in the syslog by the Hardware Event Tracker (HET). Figure 15 shows the occurrence of all errors recorded in the syslog by the HET (Figure 15a) and the errors recorded with a severity of "NON-RECOVERABLE" (Figure 15b). No HET errors were recorded between May 20 and August 23, 2019. We believe that HET errors started being recorded following a firmware update in August 2019. Based on the period for which we do have a record of HET errors, the average number of DUEs per DIMM per year is 0.00948, which yields a FIT per DIMM of approximately 1081.

4 RELATED WORK

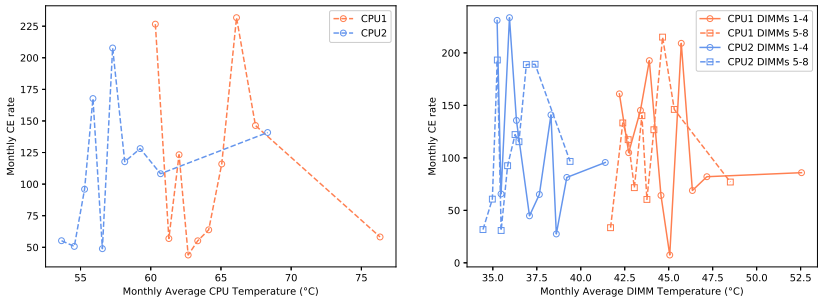
The study of failures on HPC systems has been an active research topic for over a decade [2, 5, 8, 10, 16, 17, 21, 23, 37, 38]. Failures have been studied in HPC systems [12, 27] and commercial data centers [6, 15, 20, 29]. The circumstances under which DRAM devices fail have also been studied [6, 33, 34].

Siddiqua *et al.* [32] presented a study demonstrating that the incidence of each DRAM correctable fault *mode* on the production HPC system was stable over time. Gupta *et al.* [9] studied five vastly different systems of varying sizes and hardware and software configurations to discover failure trends that are common across HPC systems. The data set covering the longest period of operation that they considered was collected on the Jaguar XT4 system from 2008-2011.

Levy *et al.* [13] examined failures over the entire lifetime of the Cielo HPC platform and showed there was no evidence of hardware aging over this interval, as might be expected. Ostrouchov *et al.* [22] examined errors and their impact on system operations for the 18,000 GPUs on the Titan system at ORNL. Lastly, Pedretti *et al.* [24] examined the software challenges of bringing up the first Petascale Arm-based supercomputer and validating its ability to run production HPC applications.

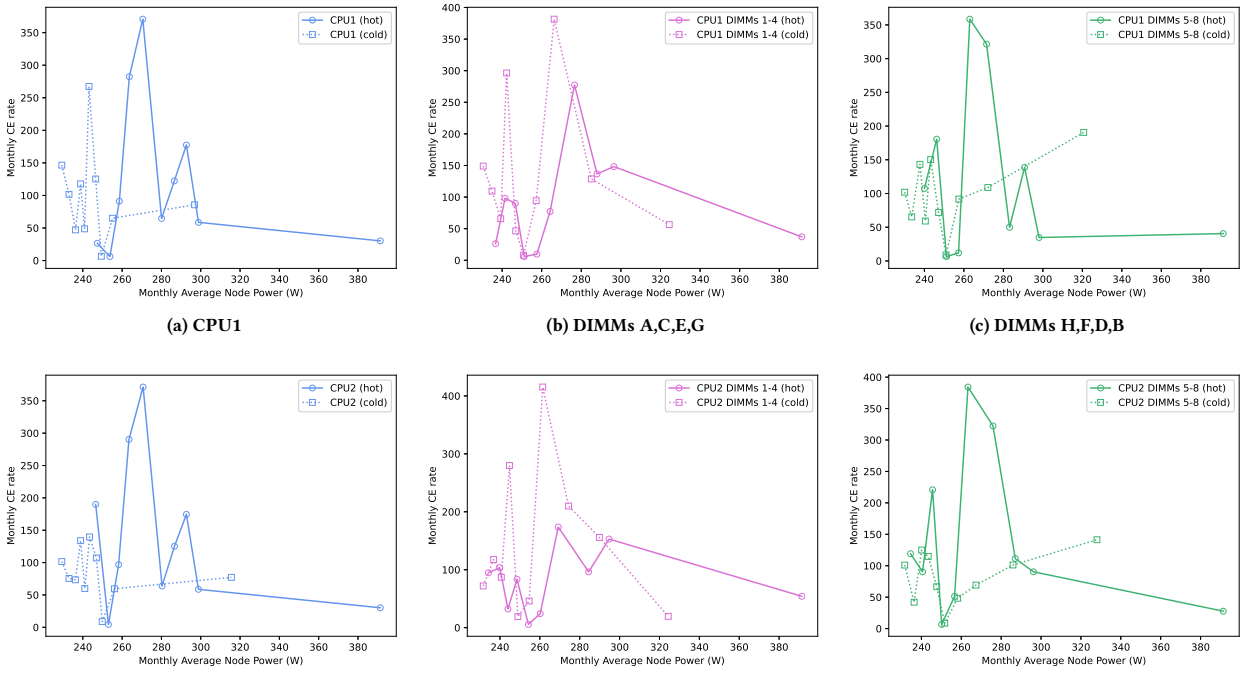
Hsu et al. [11] conclude, based on unpublished empirical data, that their data is described by the Arrehenius equation governing chemical reactions and that each 10 °C increase in temperature causes the failure rate of compute nodes to double. Sarood et al. [25] adopt this conclusion and use the Arrehenius equation to predict how much their approach is able to improve system reliability by reducing system operating temperature.

Our work is distinct from these existing studies in several important ways. First, we analyze hardware failure from the first Petascale Arm system. Second, we break-down the hardware replacement numbers during Astra's production stabilization period.



(a) CPU Temperatures **(b) DIMM Temperatures**

Figure 13: Effect of Temperature on Correctable Error Rate. These data show that on Astra there is not a strong correlation between measured device temperature and CE rate.



(a) CPU1 **(b) DIMMs A,C,E,G** **(c) DIMMs H,F,D,B**

(d) CPU2 **(e) DIMMs I,K,M,O** **(f) DIMMs J,L,N,P**

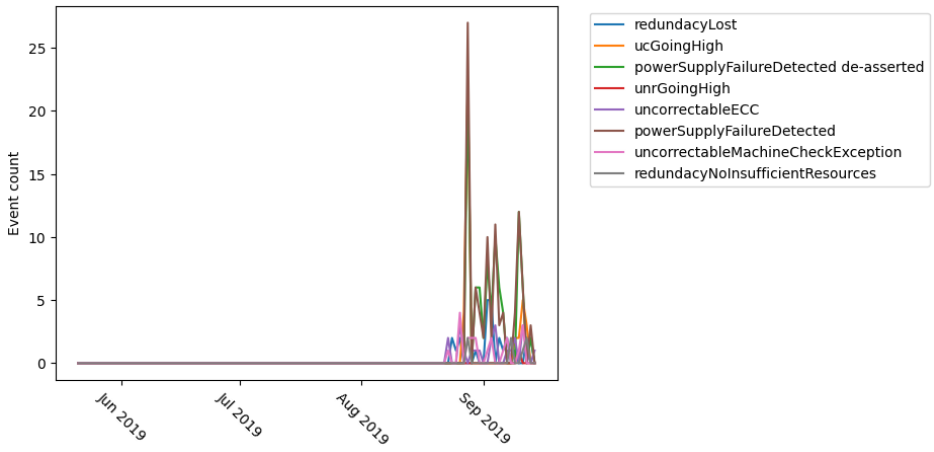
Figure 14: Effect of Utilization on Correctable Error Rate. These data show that on Astra node power (as a proxy for utilization) is not strongly correlated with correctable errors.

We also conduct a detailed analysis of CEs on Astra, detailing the failures modes and distribution of these failures across components. In addition, this work demonstrates the importance of considering faults when studying the reliability of a system and the incorrect conclusions that can be arrived at when only considering errors. Finally, we analyze uncorrectable DRAM errors.

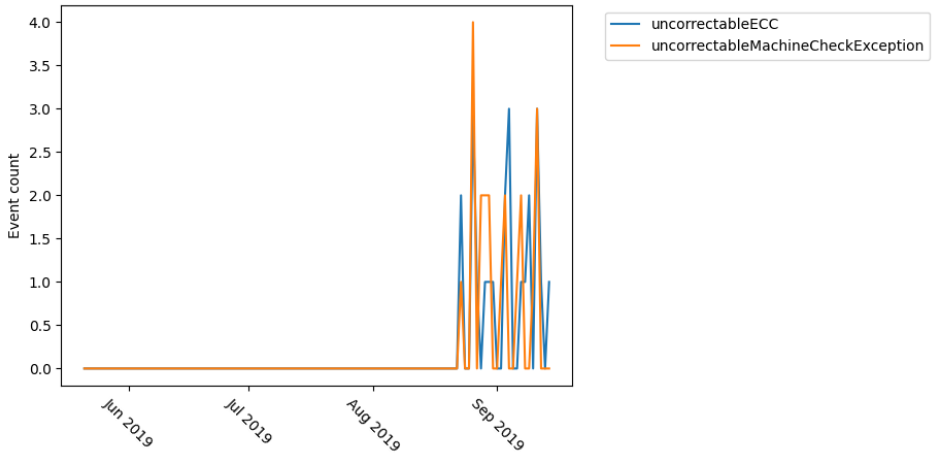
5 CONCLUSION

In this study we have provided a detailed characterization of memory-related failures on Astra, the first public Petascale, Arm-based HPC system. Specifically, we have showed the following:

- A detailed tally of the hardware components replaced in the early hardware stabilization period.
- Distinguishing between memory errors and faults on Astra gives a much clearer picture of the underlying hardware reliability and shows that faults are widely distributed throughout system-level components.



(a) All errors



(b) Non-recoverable Errors

Figure 15: Count of errors reported by the Hardware Event Tracker.

- In contrast to the existing research that found that errors occur more frequently in nodes located in the top of a rack (*see e.g.*, [10, 34]), we observed no strong correlation on Astra between a node's vertical position within a rack and the rate at which it experiences memory errors.
- Unlike the existing research showing that errors are positively correlated with temperature and utilization (*see e.g.*, [28, 30]), we observed that there was not a strong correlation between either temperature or utilization and correctable memory errors.

We believe the data and results of this analysis provide guidance to organization standing-up first-in-class platforms in HPC, organizations using Arm, and the entire HPC community in general. Therefore, the reliability and environmental data discussed in this work will be made publicly available at [7], following publication.

6 ACKNOWLEDGMENTS

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525. This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

REFERENCES

[1] A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr. Basic concepts and taxonomy of dependable and secure computing. *Dependable and Secure Computing, IEEE Transactions on*, 1(1):11–33, 2004.

[2] L. Bautista-Gomez, F. Zyulkyarov, O. Unsal, and S. McIntosh-Smith. Unprotected computing: A large-scale study of DRAM raw error rate on a supercomputer. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '16, pages 55:1–55:11, Piscataway, NJ, USA,

2016. IEEE Press.

[3] A. Clauset, C. R. Shalizi, and M. E. J. Newman. Power-law distributions in empirical data. *SIAM Rev.*, 51(4):661–703, Nov. 2009.

[4] T. J. Dell. A white paper on the benefits of chipkill-correct ECC for PC server main memory. IBM Microelectronics Division, Nov. 1997.

[5] N. El-Sayed and B. Schroeder. Reading between the lines of failure logs: Understanding how HPC systems fail. In *2013 43rd Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, pages 1–12, June 2013.

[6] N. El-Sayed, I. A. Stefanovici, G. Amvrosiadis, A. A. Hwang, and B. Schroeder. Temperature management in data centers: why some (might) like it hot. In *Proceedings of the 12th ACM SIGMETRICS/PERFORMANCE joint international conference on Measurement and Modeling of Computer Systems*, SIGMETRICS '12, pages 163–174, New York, NY, USA, 2012. ACM.

[7] K. B. Ferreira, S. Levy, J. Hemmert, and K. Pedretti. Astra memory error and system monitoring data sets. https://doi.org/10.5281/zenodo.6515019, May 2022.

[8] A. Gainaru, F. Cappello, and W. Kramer. Taming of the shrew: Modeling the normal and faulty behaviour of large-scale HPC systems. In *2012 IEEE 26th International Parallel and Distributed Processing Symposium*, pages 1168–1179, May 2012.

[9] S. Gupta, T. Patel, C. Engelmann, and D. Tiwari. 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*, SC '17, pages 44:1–44:12, New York, NY, USA, 2017. ACM.

[10] S. Gupta, D. Tiwari, C. Jantzi, J. Rogers, and D. Maxwell. Understanding and exploiting spatial properties of system failures on extreme-scale HPC systems. In *Proceedings of the 2015 45th Annual IEEE/IFIP International Conference on Dependable Systems and Networks*, DSN '15, pages 37–44, Washington, DC, USA, 2015. IEEE Computer Society.

[11] C.-H. Hsu, W.-C. Feng, and J. S. Archuleta. Towards efficient supercomputing: A quest for the right metric. In *19th IEEE International Parallel and Distributed Processing Symposium*, pages 8–pp. IEEE, 2005.

[12] A. A. Hwang, I. A. Stefanovici, and B. Schroeder. 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*, ASPLOS XVII, pages 111–122, New York, NY, USA, 2012. ACM.

[13] S. Levy, K. B. Ferreira, N. DeBardeleben, T. Siddiqua, V. Sridharan, and E. Baseman. 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.

[14] X. Li, M. C. Huang, K. Shen, and L. Chu. A realistic evaluation of memory hardware errors and software system susceptibility. In *Proceedings of the 2010 USENIX conference on USENIX annual technical conference*, USENIXATC'10, pages 6–20, Berkeley, Calif., USA, 2010. USENIX Association.

[15] X. Li, K. Shen, M. C. Huang, and L. Chu. A memory soft error measurement on production systems. In *2007 USENIX Annual Technical Conference on Proceedings of the USENIX Annual Technical Conference*, ATC'07, pages 21:1–21:6, Berkeley, Calif., USA, 2007. USENIX Association.

[16] Y. Liang, Y. Zhang, A. Sivasubramaniam, M. Jette, and R. Sahoo. Bluegene/l failure analysis and prediction models. In *International Conference on Dependable Systems and Networks (DSN'06)*, pages 425–434, June 2006.

[17] Y. Liang, Y. Zhang, A. Sivasubramaniam, R. K. Sahoo, J. Moreira, and M. Gupta. Filtering failure logs for a BlueGene/L prototype. In *2005 International Conference on Dependable Systems and Networks (DSN'05)*, pages 476–485, June 2005.

[18] K. Macarenco, K. Frye, B. Hamlin, and K. L. Karavanic. The effects of system management interrupts on multithreaded, hyper-threaded, and MPI applications. In *2016 45th International Conference on Parallel Processing Workshops (ICPPW)*, pages 338–345, Aug 2016.

[19] S. McIntosh-Smith, J. Price, T. Deakin, and A. Poenaru. A performance analysis of the first generation of hpc-optimized arm processors. *Concurrency and Computation: Practice and Experience*, 31(16):e5110, 2019. e5110 cpe.5110.

[20] J. Meza, Q. Wu, S. Kumar, and O. Mutlu. Revisiting memory errors in large-scale production data centers: Analysis and modeling of new trends from the field. In *2015 45th Annual IEEE/IFIP International Conference on Dependable Systems and Networks*, pages 415–426, June 2015.

[21] B. Nie, D. Tiwari, S. Gupta, E. Smirni, and J. H. Rogers. A large-scale study of soft-errors on GPUs in the field. In *2016 IEEE International Symposium on High Performance Computer Architecture (HPCA)*, pages 519–530, March 2016.

[22] G. Ostrouchov, D. Maxwell, R. A. Ashraf, C. Engelmann, M. Shankar, and J. H. Rogers. Gpu lifetimes on titan supercomputer: Survival analysis and reliability. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '20. IEEE Press, 2020.

[23] A. Patwari, I. Laguna, M. Schulz, and S. Bagchi. Understanding the spatial characteristics of DRAM errors in HPC clusters. In *Proceedings of the 2017 Workshop on Fault-Tolerance for HPC at Extreme Scale*, FTXS '17, pages 17–22, New York, NY, USA, 2017. ACM.

[24] K. Pedretti and et at. Chronicles of Astra: Challenges and lessons from the first petascale arm supercomputer. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '20. IEEE Press, 2020.

[25] O. Sarood, E. Meneses, and L. V. Kale. A 'cool' way of improving the reliability of HPC machines. In *Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis*, pages 1–12, 2013.

[26] M. Sato and et at. Co-design for a64fx manycore processor and "fugaku". In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '20. IEEE Press, 2020.

[27] B. Schroeder and G. A. Gibson. A large-scale study of failures in high-performance computing systems. In *Dependable Systems and Networks (DSN 2006)*, Philadelphia, PA, June 2006.

[28] B. Schroeder, E. Pinheiro, and W.-D. Weber. DRAM errors in the wild: a large-scale field study. In *Proceedings of the eleventh international joint conference on Measurement and modeling of computer systems*, SIGMETRICS '09, pages 193–204, New York, NY, USA, 2009. ACM.

[29] B. Schroeder, E. Pinheiro, and W.-D. Weber. DRAM errors in the wild: a large-scale field study. *Commun. ACM*, 54(2):100–107, Feb. 2011.

[30] B. Schroeder, E. Pinheiro, and W.-D. Weber. DRAM errors in the wild: a large-scale field study. *Communications of the ACM*, 54:100–107, February 2011.

[31] T. Siddiqua, A. Papathanasiou, A. Biswas, and S. Gurumurthi. Analysis of memory errors from large-scale field data collection. In *Silicon Errors in Logic - System Effects (SELSE), 2013 IEEE Workshop on*, 2013.

[32] T. Siddiqua, V. Sridharan, S. E. Raasch, N. DeBardeleben, K. B. Ferreira, S. Levy, E. Baseman, and Q. Guan. Lifetime memory reliability data from the field. In *2017 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT)*, pages 1–6, Oct 2017.

[33] V. Sridharan and D. Liberty. A study of DRAM failures in the field. In *Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis*, SC '12, pages 76:1–76:11, Los Alamitos, Calif., USA, 2012. IEEE Computer Society Press.

[34] V. Sridharan, J. Stearley, N. DeBardeleben, S. Blanchard, and S. Gurumurthi. Feng shui of supercomputer memory: Positional effects in DRAM and SRAM faults. In *Proceedings of SC13: International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '13, pages 22:1–22:11, New York, NY, USA, 2013. ACM.

[35] R. Stevens, J. Ramprakash, P. Messina, M. Papka, and K. Riley. Aurora: Argonne's next-generation exascale supercomputer. 3 2019.

[36] D. Tang, P. Carruthers, Z. Totari, and M. W. Shapiro. Assessment of the effect of memory page retirement on system ras against hardware faults. In *International Conference on Dependable Systems and Networks (DSN'06)*, pages 365–370, June 2006.

[37] D. Tiwari and et al. Understanding gpu errors on large-scale hpc systems and the implications for system design and operation. In *2015 IEEE 21st International Symposium on High Performance Computer Architecture (HPCA)*, pages 331–342, Feb 2015.

[38] D. Tiwari, S. Gupta, G. Gallarno, J. Rogers, and D. Maxwell. Reliability lessons learned from gpu experience with the titan supercomputer at oak ridge leadership computing facility. In *SC15: International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 1–12, Nov 2015.