

Response of HPC hardware to neutron radiation at the dawn of exascale

Andrés Bustos¹, Antonio Juan Rubio-Montero^{1*}, Roberto Méndez², Sergio Rivera^{2,3}, Francisco González², Xandra Campo², Hernán Asorey^{4,5} and Rafael Mayo-García¹

¹Department of Technology, CIEMAT, Av. Complutense 40, Madrid, 28040, Madrid, Spain.

²Ionizing Radiation Metrology Laboratory, CIEMAT, Av. Complutense 40, Madrid, 28040, Madrid, Spain.

³Nuclear Engineering Department, Universidad Politécnica de Madrid, José Gutiérrez Abascal 2, Madrid, 28006, Madrid, Spain.

⁴Medical Physics Department, Comisión Nacional de Energía Atómica, Centro Atómico Bariloche, Av. E. Bustillo 9500, San Carlos de Bariloche, 8400, Argentina.

⁵Instituto de Tecnologías en Detección y Astropartículas (ITeDA), Comisión Nacional de Energía Atómica, Centro Atómico Constituyentes, Av. Gral. Paz 1499, Villa Maipú, 1650, Argentina.

Contributing authors: antonio.rubio@ciemat.es;

Abstract

Every computation presents a small chance that an unexpected phenomenon ruins or modifies its output. Computers are prone to errors that, although may be very unlikely, are hard, expensive or simply impossible to avoid. In the exascale, with thousands of processors involved in a single computation, those errors are especially harmful because they can corrupt or distort the results, wasting human and material resources. In the present work, we study the effect of ionizing radiation on several pieces of commercial hardware, very common in modern supercomputers. Aiming to reproduce the natural radiation that could arise, CPUs (Xeon, EPYC) and GPUs (A100, V100, T4) are subject to a known flux of neutrons coming from two radioactive sources, namely ²⁵²Cf and ²⁴¹Am-Be,

in a special irradiation facility. The working hardware is irradiated under supervision to quantify any appearing error. Once the hardware response is characterised, we are able to scale down the radiation intensity and to estimate the effects on standard data centres. This can help administrators and researchers to develop their contingency plans and protocols.

Keywords: Exascale; silent errors; neutron radiation; atmospheric radiation; induced failures; HPC.

1 Introduction

In modern computations, massive parallelism is ubiquitous when the application allows it. It saves time and allows computations that would have been unfeasible otherwise. In spite of all the advantages that this presents, the higher the number of processes/threads involved, the larger the probability of one of them failing and affecting the computation. Cosmic ray radiation is one of the possible sources of such errors. When a high energy particle interacts with the atmosphere, it creates a cascade of secondary particles that reaches the Earth's surface. Among those particles, neutrons are susceptible to interact with the hardware, flipping one or more bits in the registers and perturbing the computation. Thus, it induces a probability of occurrence of an error depending on the energy and the flux of the incoming neutrons as well as on the density of processors per volume. There are several types of such failures depending on their after effects. The most undesirable case is known as Silent Error (SE), which corrupts the computation output in an undetectable manner. As a consequence, software and hardware resilience to radiation induced errors is a key issue in computing science towards the exascale. More restrictive or relaxed contingency plans to overcome those errors can be implemented depending on the Mean Time Between Failures (*MTBF*) of the hardware components.

In the present work, we perform a series of experiments devoted to studying the response of commercial hardware devices to a neutron flux originated by radioactive materials. The Neutron Standards Laboratory (LPN) is the Spanish national reference in neutron metrology and it is one of the installations that constitute the Ionizing Radiation Metrology Laboratory (LMRI) at CIEMAT. LPN counts with two neutron standards based on ²⁵²Cf and ²⁴¹Am-Be neutron sources, currently used for calibration purposes. These sources provide well-known neutron flows that can be used to irradiate material or devices. In our experiments, we will irradiate computing nodes with various CPUs and GPUs.

Checkpointing and rollback recovery are the *de-facto* general-purpose error recovery techniques. They employ checkpoints to save periodically the state of a parallel application so that when an error strikes some process, the application can be restored to one of its former states. Nevertheless, the problem can be approximated from different perspectives as well.

Thus, the work carried out so far on characterizing the cosmic radiation on computing processors has basically focused on demonstrating their effect to be translated to silent errors and proposing solutions (L1 bypass, metal shielding, etc.). They have also produced noticeable results in characterizing the radiation effects on different computing units (memories, processors, or modern accelerators), but with a huge neutron flux if compared with the actual cosmic radiation received at the sea level (around six orders of magnitude higher in the referenced GPUs and Xeon Phi's studies, see for example [1, 2]). Then, in order to better determine the actual effect of cosmic rays is necessary to correlate the dose received to the flux that happens at any latitude, longitude, and altitude around the world with the actual experimental rate of SE per neutron flux.

This paper is organised as follows: in Sec. 2 we briefly review the scientific work regarding silent errors and hardware irradiation; in Sec. 3 we describe the irradiation facility and numerical tests employed; in Sec. 4 we discuss the results and finally in Sec. 5 we present our conclusions.

2 Related Work

2.1 Related work on overcoming silent errors

Silent data corruption has been typically studied by comparing the experimental output with the expected result [3]. In order to find a suitable solution, it is needed to consider error criticality, this is, the impact of this corruption on the application or system (see initial results in [1]).

Considerable efforts have been directed to reveal silent errors. In [4], a comprehensive list of techniques and references can be found. Most of the current techniques have combined redundancy at various levels, together with a variety of verification mechanisms. The classic approach is at the hardware level, where all computations were executed twice or even in triplicate, and majority voting was enforced in case of different results [5]. Another hardware-based error detection approach is proposed by Moody [6] and the use of ECC (Error Correcting Code) memory, which can detect and even correct a fraction of errors, but in practice it is complemented with software techniques because not all parts of the system are ECC-protected (in particular, logic units and registers inside the processing units). Then, it clearly rises up that a more advanced integration between these two methods must be accomplished.

Regarding verification mechanisms via application specific information, they include memory scrubbing [7], fault-tolerant algorithms [8], Algorithm Based Fault Tolerance (ABFT) techniques [9], coding sparse matrix-vector multiplication kernels [10], coupling a higher-order with a lower-order scheme for ordinary differential equations [11] and critical MPI message validation [12].

A set of novel silent data corruption detectors by leveraging support vector machine regression have been explored in [13]. Attempts have also been made to use genetic algorithms to detect such errors. Such detectors have demonstrated high precision and recall, but only if they run immediately after an

error has been injected. A neural network detector that can identify silent data corruptions even multiple iterations after they were injected is proposed in [14].

In the field of sparse systems and for the sake of completeness, it is of interest to consider the preconditioned conjugate gradient method proposed by Chen [15] to work with sparse iterative solvers. Another iterative algorithm which deserves consideration is GMRES, which has been incipiently analysed with soft fault error models [16]. Other articles have focused on evaluating algorithms for both stability and accuracy in the presence of faults [17], detecting software errors by using properties of the algorithm [18], or providing more information and control for the library or application in handling likely errors, such as different types of DRAM failures [8, 19].

Between generic and application oriented verification mechanisms, approximate replication should be referenced. This methodology can be applied either at numerical method [11] or hardware [20] levels by comparing the exact floating-point results with the ones of an approximate operator. It has shown promising results, but again it is still not general enough, since each application needs to be manually complemented with the required computing kernels. Then, transparent and agnostic developments should be designed and [21] has worked in such a direction by combining an analytical model replication and checkpointing.

Regarding studies on optimal checkpointing and/or verification periods, two models have been mostly used: (i) errors are detected after a certain delay following a probability distribution (typically, an exponential distribution); (ii) errors are detected through some verification mechanism [22]. In both cases, the optimal period in order to minimise the dead time can be computed, i.e., the fraction of time where nodes do not perform useful computations. In practice, only a fixed number of checkpoints can be kept in memory, and the first model may lead to an irrecoverable failure. In this case, the solution has been to determine the period required for an acceptable risk. For the second model, there is no risk of irrecoverable failure, due to the verification mechanism. The right replication level to detect and correct silent errors at scale is studied in [23]. A more realistic model assuming that errors are detected through some verification / validation mechanism (re-computation, checksums, coherence, etc.) would be an improvement of the overhead induced in this kind of solution. See for example [24], in which 7-70 percent less overhead than a full duplication technique with similar detection recall is achieved.

Traditionally, a general assumption for building up a resilient recovery system has been to consider that each checkpoint forms a consistent recovery line and silent errors strike according to a Poisson process. Additionally application workflows refer to a number of parallel tasks that exchange data at the end of their execution, i.e. the task graph is a linear chain, and each task (except maybe the first and the last one) reads data from its predecessor and produces data for its successor. Bearing this in mind, there exist results describing the efficiency of copying with silent errors by combining checkpointing with some verification mechanism [25]. Thus, it is possible to design a general-purpose

technique based upon computational patterns that periodically repeat over time, i.e. optimise the aforementioned trade-off between error-free overhead and execution time. These patterns interleave verifications and checkpoints, so a pattern minimizing the expected execution time can be determined. From this point on, it is possible to move to application-specific techniques via dynamic programming algorithms for linear chains of tasks as well as ABFT-oriented algorithms for iterative methods in sparse linear algebra. We can see promising results in [26] with a sparse grid combination technique applied to several scientific fields. In the case of an application composed of a chain of tasks, the optimal and dynamic checkpointing strategy has been solved in [27], but there is still room for designing new algorithms exploiting a proper verification technique as well as for integrating multi-level checkpointing in order to cope with both fail-stop and silent error. The User Level Failure Mitigation (ULFM) interface enables the implementation of resilient MPI applications, system runtimes, and programming language constructs by detecting and reacting to failures without aborting their execution [28].

Regarding ABFT algorithms, the (preconditioned) Conjugate Gradient and GMRES methods can be used for both detection and correction [29]. But it can be extrapolated to any iterative solver that uses sparse matrix vector multiplications and vector operations (non-stationary iterative solvers such as CGNE, BiCG, BiCGstab, etc.). Also, ABFT allows detecting soft errors in the LU Decomposition with Partial Pivoting (LUPP) algorithm [30], a method widely used but with serious limits in scalability.

Recent compilation studies on fault tolerance in exascale systems can be found in [31] and [32].

2.2 Related work on radiating hardware

To date, several articles have been published about the effect of radiation on computing hardware. Initially, it has been studied by Ziegler et al. in the late '90s [33] by irradiating different 16-MB DRAM memory. They concluded that silent errors induced by cosmic radiation depend on the memories' cell technology used. In order to demonstrate to the manufacturers that the errors appearing in ASCI Q at Los Alamos Nat. Lab. were produced by cosmic rays, one of the servers was placed in front of a beam of neutrons causing errors to spike. This evidence was complemented by studies devoted to radiation-induced soft errors in advanced semiconductor technologies [34] and neutron induced single event upset dependence on bias voltage for CMOS SRAM [35].

Other experimental results related to the occurrence of silent errors are the single-bit ECC errors rate of 350 min^{-1} and the double-bit errors once per day both observed in the Jaguar supercomputer in 2006. The latter error class was detected, but not corrected by ECC techniques.

Radioactive lead in the solder to cause bad data in the L1 cache was observed in BlueGene/L at Lawrence Livermore Nat. Lab., a fact that derived in the necessity of bypassing L1 and, consequently, slower computations. Also, the U.S. Department of Energy's Titan has a radiation-induced *MTBF* in

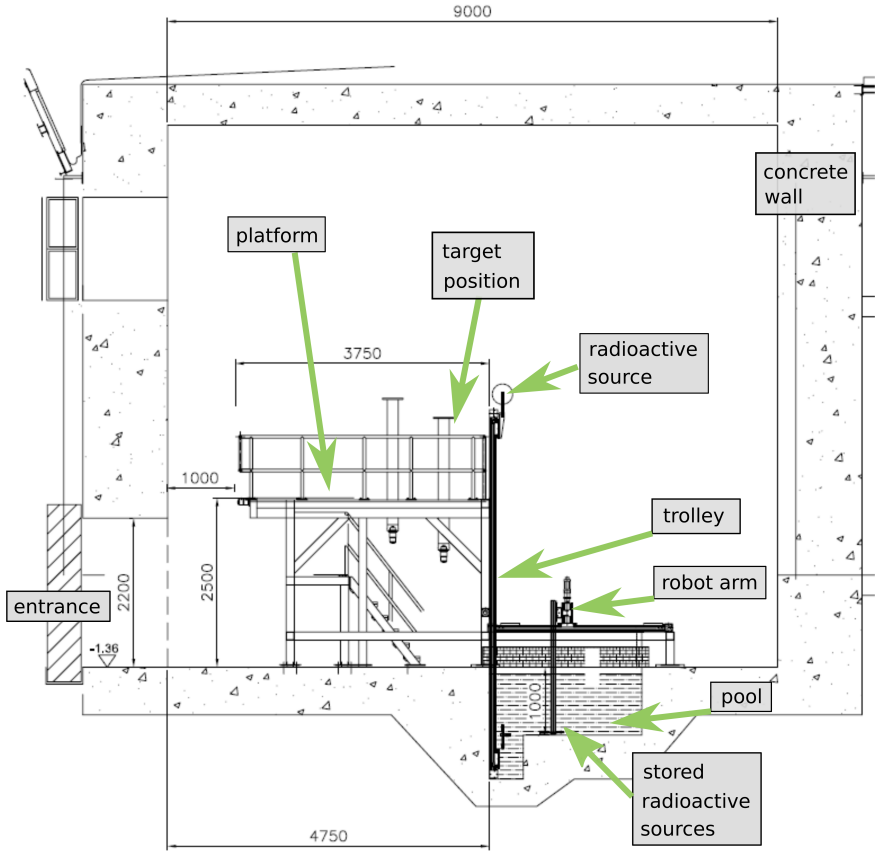
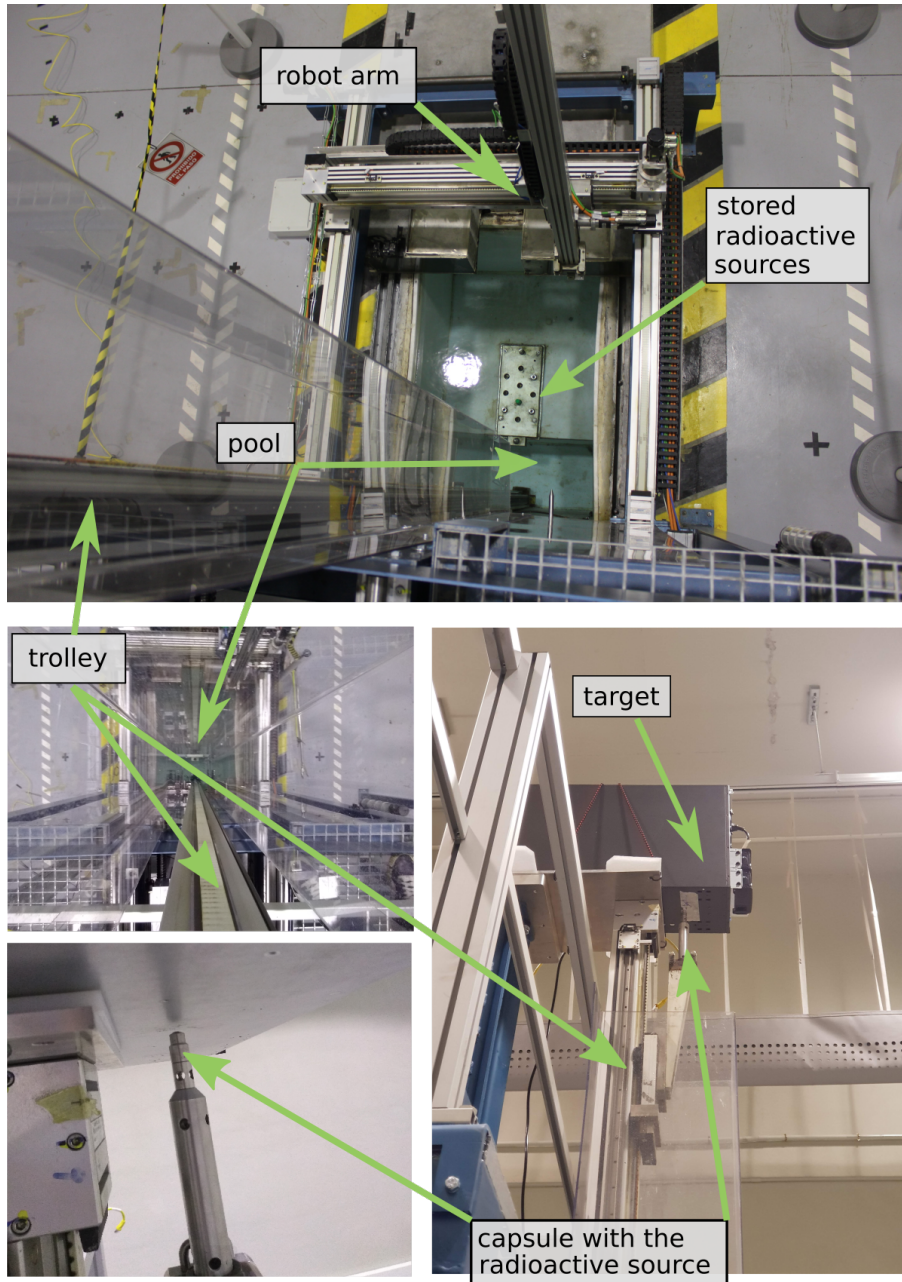


Fig. 1 Scheme of the experimental facility. Distances are expressed in millimetres.

the order of dozens of hours in their Kepler GPUs [36]. Hence, works elaborating on the direct dependence between silent errors and radiation have been published focused on determining the reliability in GPUs [1] and Xeon Phi, also applying high-level fault injection [2]. These works also quantify and qualify radiation effects on applications' output by correlating the number of corrupted elements with their spatial locality and providing the mean relative error (dataset-wise) to evaluate radiation-induced error magnitude. Comparison between high-energy and thermal neutrons effects on processor and memory error rates has been studied as well [19].

**Fig. 2** Pictures of the experimental set-up.

3 Experimental Set-Up

3.1 Irradiation Facility

In this work, the Neutron Standards Laboratory (LPN) has been employed as irradiation facility to study the response of hardware devices to neutron fluence [37].

LPN counts with an irradiation room, that is a bunker with dimensions of 7 m x 9 m x 8 m, following ISO 8529-1 recommendations [38] and with 1.25 m thick walls, as it is shown in Fig. 1 [37]. The calibration neutron sources are stored in water that works as a very efficient neutron shield. They are remotely manipulated making use of a Cartesian manipulator, and a launcher, to select the neutron source and move from its storage position in the bottom of a pool to the irradiation position, 4 m over the ground. In front of the neutron source, the equipment to be calibrated is placed, over an automated table that allows precise positioning at the desired height and distance. These systems are controlled from a Control Room outside the bunker. Some pictures of the facility are shown in Fig. 2.

The neutron sources used in the experiments are ^{252}Cf and $^{241}\text{Am-Be}$, which are currently employed in routine calibrations. Both sources provide neutron spectra in the fast energy range, see Fig. 3, according to ISO 8529-1 recommendations [38]. First, the device to be irradiated is placed on the automated table. Then, using the remote systems the neutron source is placed under this device, as seen in Fig. 2. ^{252}Cf is placed inside a cylindrical capsule with external dimensions of 9.8 mm (H) x 7.9 mm ($\varnothing$). This capsule is itself inside another capsule holder, handled by the automated manipulation system (Fig. 2).

The Cf neutron source consists of 236 mg of ^{252}Cf and according to the calibration certificate, the emission rate was $B_{\text{Cf0}} = 5.471 \times 10^8 \text{ s}^{-1} \pm 2.6\% (2\sigma)$, measured by the National Institute of Standards and Technology (NIST, USA), on 12/05/2012. From this value, and knowing that the half-life for ^{252}Cf is $T_{1/2} = 2.65$ years, and assuming an exponential decay, the current emission rate can be calculated for the experiments dates of Subsections 4.1 4.2, 4.3 and 4.4.

On the other hand, the $^{241}\text{Am-Be}$ neutron source consists of a compacted mixture of $^{241}\text{AmO}_2$ and ^9Be powder doubly encapsulated. This source has a nominal activity of 185 GBq and emits $1.11 \times 10^7 \text{ s}^{-1} \pm 1.4\% (2\sigma)$ neutrons, traceable to the Czech Metrology Institute (CMI) on 1/1/2012. The capsule has dimensions of 48.6 mm (H) x 19.1 mm ($\varnothing$) and it is inside its own capsule holder. Considering the value of $T_{1/2} = 432.6$ years for $^{241}\text{Am-Be}$, the current emission rate can be determined for the irradiation dates, being this value practically constant for the different irradiation campaigns, and very close to the original value, $B_{\text{AmBe}} = 1.09 \times 10^7 \text{ s}^{-1}$.

The irradiation distance is defined as the distance between the end of the capsule holder and the external surface of the server chassis. Because of safety issues, the smallest irradiation distance is 0.5 cm. This is not the real distance

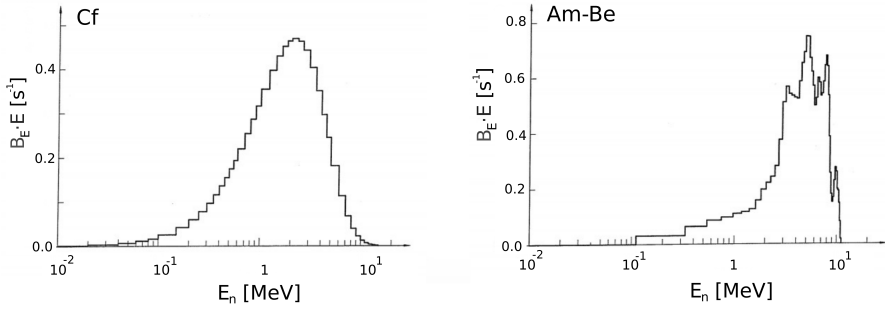


Fig. 3 Neutron energy spectra for both sources. Note that the ^{252}Cf peaks at 2 – 3 MeV and the $^{241}\text{Am-Be}$ at slightly higher energies.

between the source and the irradiated device, because it is necessary to take into account the capsule holder thickness (4 mm), the distance to the centre of the source and the distance from the external chassis of the server to the target device inside.

In order to have a more realistic estimation of this distance (d) between the radioactive source and the target hardware, we introduce a correction offset to the distance set by the remote system. We add the capsule holder correction (~ 0.5 cm) and a half of the capsule size. This leads to an offset of +1.0 cm and +3.0 cm to the Cf and Am-Be distances, respectively. In the experiments, the source was placed perpendicularly and centred on the surface of the target chip to irradiate. The distances inside the server were previously measured with calliper. However, we also assume that the corrected distance has an uncertainty of ± 0.5 cm, used in later error calculations in Sec. 4.

In the experiments of Sec. 4 we may be able to provide a rough estimation of the effective cross section of the hardware piece. In this sense, we are able to measure the error rate E dividing the number of observed errors by the time lapse, and then:

$$\sigma_{\text{eff}} \approx \frac{E \times 4\pi d^2}{B_i}, \quad (1)$$

being B_i is the neutron emission rate (subscript i labels the source) and $B_i/(4\pi d^2)$ is the flux (neutrons per surface per time unit).

3.2 Hardware description

The hardware irradiated in this work is standard in many HPC data centres. We are provided with two modern computing servers, with two different CPU models and with three GPUs relying on different lithography technologies, whose transistor sizes range from 7 nm to 14 nm. We have an Intel Xeon CLX Gold 6230 Cascade Lake, 20c (2.1 GHz) and an AMD EPYC Rome 7282, 16c (2.8 GHz). In terms of GPUs, we have a T4, a Tesla V100 16GB and an A100 40GB, all manufactured by the NVIDIA Corp., but based on different microarchitectures, namely Turing, Volta and Ampere, respectively.

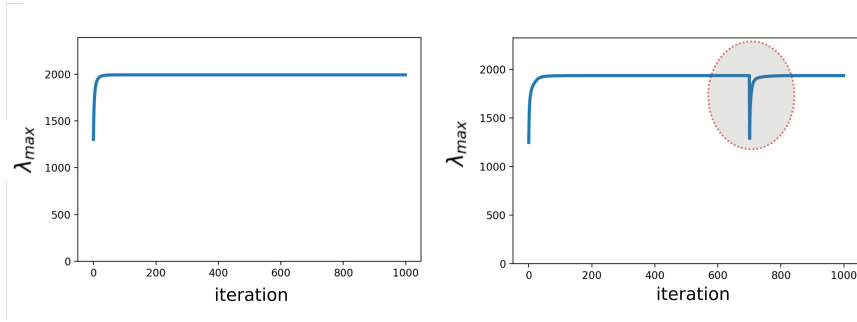


Fig. 4 Executions of the power method: left, a reference execution; right, an execution artificially perturbed in a single value of the eigenvector.

As preparation for every experiment, we removed the unnecessary hardware from the servers. Thus, the RAM memory modules were reduced to the minimum and a single hard disk is used, keeping only the components strictly needed for running the OS and the programs. In this way, we minimise any possible effect induced in those components by the neutron radiation. We also shield, when possible, the parts of the server that should not receive neutrons with 1 cm width polyethylene tiles, specially the hard disk.

3.3 Numerical Experiments

Since we are irradiating both CPUs and GPUs, we prepare different numerical experiments for each case. The computation itself is irrelevant, but we design it so the CPU/GPU is running at full capacity. The output of those experiments is monitored during the irradiation to detect any errors.

A basic CENTOS 7.9 Linux operative system is installed in each computing node. On top of that, Python is installed via Anaconda. CUDA 11.6 and cuDNN 8 libraries are installed too in the nodes to use the GPUs for scientific computing.

For the CPU experiments, we choose to use the so-called power method for finding the largest eigenvalue of a symmetric matrix [39]. This is a simple and iterative procedure, and if we fix the random seed then we can exactly reproduce the computation and track numerical errors during the irradiation. We can modify the size of the matrix and the number of iterations to control memory usage and execution time. Fig. 4 shows a reference execution, run previously to the experiments, and another execution where we artificially introduce some random noise. This procedure allows the identification of any anomaly in the computation, either looking at the values of the eigenvalue as a function of the iterations, or computing the root mean square value [RMS] of both time series. If this number is greater than the machine error (in this case $\sim 10^{-16}$), then we have the indication of an unexpected modification of a numerical variable in the hardware.

In the case of GPUs, we take advantage of their huge parallelisation capabilities and train a deep neural network image classifier. We use the well-known

Fashion MNIST dataset [40]: a set of 60,000 greyscale images of fashion articles (handbags, shoes, shirts, etc.), with 28x28 pixel resolution. We code a convolutional neural network (CNN) classifier and train this model with the whole dataset. The model is composed of several CNN and Maxpool layers, with a final Softmax layer that performs the classification. The number of layers and neurons in each layer is adjusted depending on the GPU memory to the maximum size available. Additionally, we use a batch size of 60,000 (the whole dataset) so the GPU works at full capacity. The GPU usage level is checked using the `nvidia-smi` tool. Considering that the neural network training produces a time series of the loss function values, we can detect SE in the same way as in the CPU tests.

Both numerical experiments are tuned to last approximately five minutes, and the computations are cyclically running during the irradiation experiments. When the experiment is over then the neutron source is returned back to the safety pool. We can check the outputs and OS kernel messages and quantify and classify any error. Knowing the irradiation time we can estimate the *MTBF* of the target piece of hardware. This experimental procedure requires constant supervision, since some errors can freeze the OS or even they can reboot the computer and it could require human intervention.

We will differentiate three categories of errors according to their effects on the irradiated hardware from the application point of view:

- **Auto-corrected Errors (AE):** those errors appear and they are immediately corrected by the computing node itself using ECC techniques. They also are harmless and they do not modify the computation output in any aspect. AE can be detected either by a message in the application terminal or in the system logs.
- **Catastrophic Errors (CE):** in this case, the neutron hits a critical component of the hardware in an irrecoverable way. Then the computation is interrupted, and maybe the OS deactivates the hardware component or even the system is blocked and must be rebooted. In any case, CE are easy to identify and require manual intervention by the user. Notice that in a parallel job, a single CE in one of the nodes can stop the whole computation, with the consequent resource waste.
- **Silent Errors (SE):** as we mentioned in the Introduction, this error category is by far the most dangerous. Silent errors are not detected by the OS nor have any detectable impact on the hardware, but they change the computation output. Thus they modify the results in an undetectable way and they compromise the goodness of the outcome. In some cases, this silent data corruption can be easily identified *a posteriori* by the user because they lead to unrealistic results. But in some other cases, the output is plausible, simply giving misleading values. Consequently, SE must be avoided by all means. An example of a SE would be, in a weather forecast code, an increment of 20 km/h in the wind speed prediction. Decisions based on this data can needlessly trigger hurricane protocols and alerts. Or even worse, if the variation is -20 km/h, a hurricane forecast can be unnoticed with

all its consequences. Many other cases, such as autonomous cars or aerospace engineering, are also very sensitive to SE.

Because the most common natural cause of the SE is the cosmic neutron radiation, this work is focused on experimentally characterizing the *MTBF* corresponding to different common processors used in supercomputers (both CPUs and GPUs) as a dependence of the incoming radiation. Thus, the experimental results obtained will complement the previous estimations of the natural radiation on diverse data centres that currently are hosting exascale computers [41].

4 Results & Discussion

In this section we present the experimental results obtained with the previously discussed set-up. We divide the results into four groups, each one corresponding to a ~ 2 week campaign of the radiation facility, from November 2020 to April 2022. Within each campaign, we consider the emission rate of the Cf source constant. Recall that the Am-Be emission rate is considered as constant over all campaigns due to its long lifetime.

The numerical data can be found in Table 1 for all the hardware considered. The uncertainties on the counts are estimated assuming that the events follow a Poisson distribution and applying quadratic propagation of errors. In the following subsections we will discuss every case separately.

4.1 INTEL Xeon Gold 6230 CPU

In the experiments with this processor, we have observed many AE logged in the system, being the Cf emission rate of $B_{\text{Cf}} = 5.48 \times 10^7 \text{ s}^{-1}$. We can see in Table 1 that there is a clear positive correlation between the *MTBF* and the distance to the source. In Fig. 5 we plot the $MTBF_{\text{AE}}$ as a function of the distance from the irradiated computer to the radioactive source (Cf or Am-Be). Note that, because the number of events (AE, CE, SE) in Table 1 is not very large, the errors on the magnitudes are important, being 10-30 %.

We fit the $MTBF_{\text{AE}}$ to a function $MTBF_{\text{AE}}(d) = A \times (d+B)^2 + C$, taking into account the uncertainties in $MTBF_{\text{AE}}$. This function is chosen because the radiation intensity scales with d^{-2} for a perfect isotropic and punctual source, and consequently $MTBF_{\text{AE}} \sim d^2$. Here A plays the role of a constant that depends on the source, B represents a constant offset to the distance and C represents contributions from external sources, such as background radiation. Given the values of χ^2 divided by the degrees of freedom, see Fig. 5, we can conclude that the data can be coarsely described by the parabola we defined, given the low statistics in the points at higher distances. The shaded area in Fig. 5 represents the error intervals of the fits, calculated using quadratic error propagation with the standard deviations of the fit parameters. It is clear then that many other functional dependencies of $MTBF_{\text{AE}}$ and d are also compatible with the experimental data.

Hardware	Source	d [± 0.5 cm]	t [min]	AE	$MTBF_{AE}$ [min]	CE	$MTBF_{CE}$ [min]	SE
Intel	Cf	2.5	437	181	2.4(2)	1	437	0
Xeon	Cf	3.2	260	66	3.9(5)	1	260	0
Gold	Cf	4.0	337	93	3.6(4)	2	$17(12) \times 10^1$	0
(Cascade	Cf	7.0	178	42	4.2(7)	0	-	0
Lake)	Cf	10.0	178	16	11(3)	1	178	0
[14 nm]	Cf	14.0	384	14	27(7)	0	-	0
	Cf	18.0	357	11	32(10)	0	-	0
	Am-Be	4.5	346	35	10(2)	0	-	0
	Am-Be	5.0	326	34	10(2)	0	-	0
	Am-Be	6.0	390	25	16(3)	0	-	0
	Am-Be	7.5	322	12	27(8)	1	322	0
AMD EPYC (Rome) [7 nm]	Cf	2.1	2134	239	8.9(6)	2	$11(8) \times 10^2$	0
NVIDIA	Cf	3.8	541	0	-	24	23(5)	9*
A100	Cf	4.6	269	0	-	12	22(7)	5*
(Ampere)	Cf	6.4	380	0	-	7	54(20)	7*
[7 nm]	Am-Be	5.5	1140	0	-	20	57(13)	16*
	Am-Be	8.2	797	0	-	8	$10(4) \times 10^1$	8*
NVIDIA	Cf	5.2	415	0	-	0	-	0
V100	Am-Be	7.2	385	0	-	1	385	0
(Volta) [12 nm]								
NVIDIA	Cf	5.2	468	0	-	34	13(2)	0
T4	Cf	6.2	520	0	-	21	25(5)	0
(Turing)	Am-Be	7.2	420	0	-	4	$10(5) \times 10^1$	0
[12 nm]	Am-Be	8.2	420	0	-	3	$14(8) \times 10^1$	0

Table 1 Results of the irradiation experiments on five pieces of hardware using two radioactive sources. d is the distance to the source and t is the irradiation time. AE stands for Auto-corrected Error, CE for Catastrophic Error and SE for Silent Error (see Sec. 1). All times (t , $MTBF_{AE}$, $MTBF_{CE}$) are measured in minutes. The uncertainty estimation is indicated in parenthesis when the number of counts is larger than one. The only SE candidates (marked with an asterisk) are discussed in detail in Sec. 4.3. Each row of the table may contain aggregated data from experiments performed on different days.

The effective cross section, see Eq. 1 is depicted in Fig. 6. Given the large uncertainties observed, a dependence of the cross section with the distance to the source still remains, but it can be considered approximately constant within the obtained errors for $d \gtrsim 8$ cm when the Cf were used. In any case, we may estimate this cross section to be of the order of $3 - 5 \times 10^{-8} \text{ cm}^2$ for both sources. Although the Am-Be source produces a smaller number of errors, its lower emission rate compensates and leads to a cross section similar to the Cf.

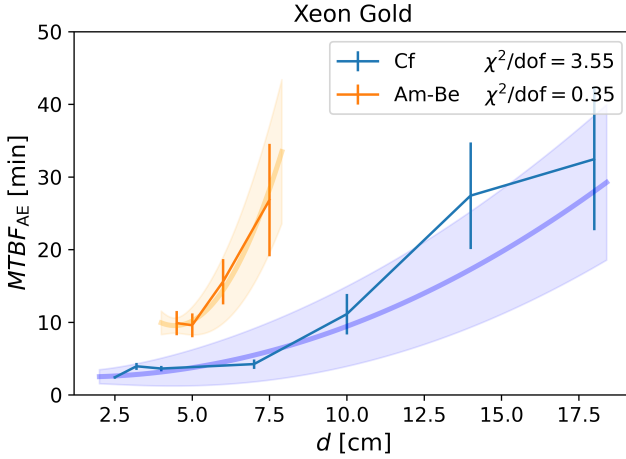


Fig. 5 *MTBF* for the AE error class in the Xeon Gold processor and its fit to a parabola.

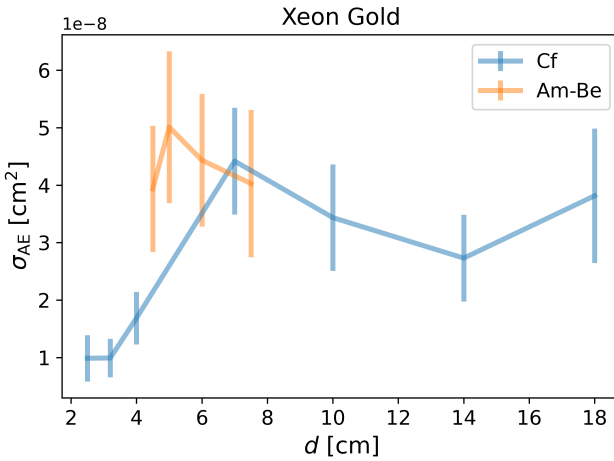


Fig. 6 Effective cross section for the AE error class in the Xeon Gold processor.

Regarding the CE, we have observed a very small amount of them. Their rate is very small, typically one every few hours, thus there are not enough statistics to find any tendency as we did with the AE.

4.2 AMD 7282 EPYC Rome CPU

This CPU presents the best inherent resistance to radiation of all the tested devices. According to Table 1, the AMD CPU presents an AE error rate of $\approx 0.011(7)$ errors/min and a CE error rate of $0.0009(7)$ errors/min. In terms of *MTBF* we have $MTBF_{AE} = 8.9(6)$ min and $MTBF_{CE} = 11(8) \times 10^2$ min. The corresponding effective cross sections are $\sigma_{AE} = 2.5(1.2) \times 10^{-9} \text{cm}^2$ and

$\sigma_{\text{CE}} = 2.1(1.8) \times 10^{-11} \text{cm}^2$. Due to the fact that the CE error rates are very small and the time limitations of the LPN (it is important to highlight that this is the neutron standard facility in Spain and must provide continuous service to the domestic community), we do not make experiments irradiating at larger distances or with the Am-Be source. The Cf activity is lower than in the previous section, being $B_{\text{Cf}} = 4.13 \times 10^7 \text{s}^{-1}$. Still, σ_{CE} for this processor is orders of magnitude smaller than in other hardware, and almost compatible with zero within its error bars due to the small number of events measured. Note that although this CPU was launched on similar date than the aforementioned Xeon processor, the lithography technology is completely different. AMD is encapsulating the computing cores in separated *chiplets* while reducing the transistor size to a half (7 nm) with exception of the E/S manager (14 nm).

4.3 NVIDIA A100 GPU

These experiments are performed with a Cf emission rate of $B_{\text{Cf}} = 4.22 \times 10^7 \text{s}^{-1}$. Similar to the other GPUs (see Sec. 4.4), no AE are observed in the system logs or on the terminal while executing the neural network training. Only CE and indications of SE are encountered.

In Fig. 7 we plot the $MTBF_{\text{CE}}$ as a function of the distance from the irradiated computer to the radioactive source (Cf or Am-Be). For short distances, we find an $MTBF_{\text{CE}}$ of roughly 20-30 min using Cf and larger than one hour using Am-Be. In Fig. 8 we observe that the effective cross section in this case is $\approx 10^{-8} \text{cm}^2$.

Silent errors are detected by comparing the RMS of the learning curves (loss function value vs training time) of the CNN training and a reference run. If nothing disturbs the training, the RMS value is of the order of machine precision ($O(10^{-16})$). When a SE occurs, this value is larger. We have detected several dozens of SE that give an RMS of $O(10^{-2})$. In Fig. 9 we plot the fractions of silent errors over the total number of runs in each case of Table 1. We also run some extra cases executed without radiation. We can see that all the values are essentially compatible with each other within their uncertainties. This indicates that the observed SE were not triggered by the radiation. However, since the experiments without radiation were the last ones, we cannot discard that the radiation caused permanent damage to the A100 at the beginning of the campaign.

4.4 NVIDIA V100 and T4 GPUs

The V100 GPU presents very strong resistance to incoming neutrons. As we can see in Table 1, having the neutron source as close as possible to the target produces very few events. No AE nor SE appeared, and only one CE occurred in the GPU when using Am-Be. The differences between the two sources can be attributed to their distinct activities and energy spectra (Fig. 3) The Cf emission rate is the largest among all our experiments: $B_{\text{Cf}} = 5.97 \times 10^7 \text{s}^{-1}$.

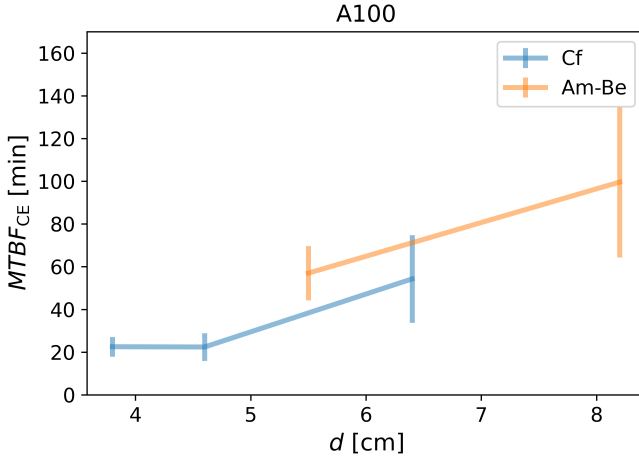


Fig. 7 $MTBF$ for the CE error class in the A100 GPU.

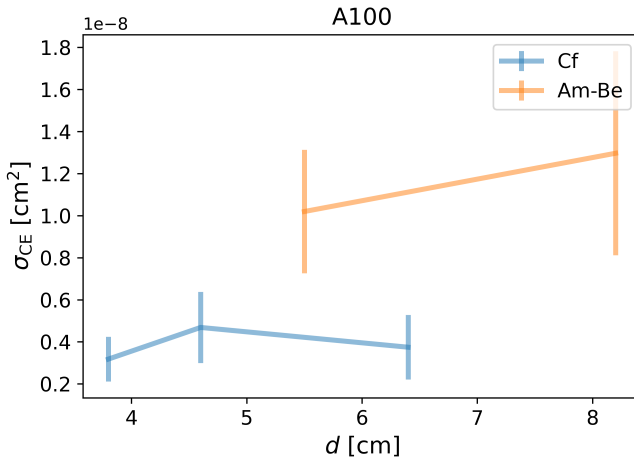


Fig. 8 Effective cross section for the CE error class in the A100 GPU.

On the other hand, the T4 GPU suffered CE more often, specially with Cf. We detected a CE every 13 or 25 minutes, depending on the distance to the Cf source. The Am-Be source induces an $MTBF_{CE}$ one order of magnitude higher. Table. 2 shows the estimation of the effective cross section (Eq. 1). Together with the absence of SE, it indicates that checkpointing techniques are appropriate in this radiation scenario.

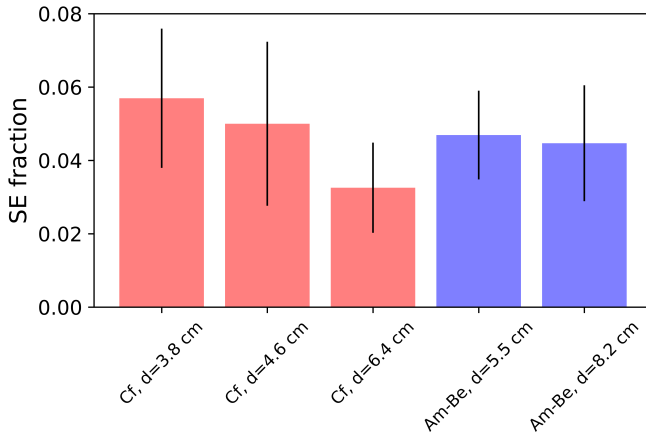


Fig. 9 Fraction of SE detected in the A100 GPU. The vertical black lines over each bar represents the statistical error of the measurements.

Source	$d [\pm 0.5 \text{ cm}]$	$\sigma_{\text{CE}} [\text{cm}^2]$
Cf	5.2	$7(2) \times 10^{-9}$
	6.2	$5.4(1.5) \times 10^{-9}$
Am-Be	7.2	$9(5) \times 10^{-9}$
	8.2	$9(5) \times 10^{-9}$

Table 2 Effective cross section estimation for the T4 GPU. In the case of Am-Be source, both cross sections are indistinguishable because of the error bars.

4.5 Scaling down the measurements to an estimation of errors in operation status

Considering the experiments documented in the previous subsections, we will focus on the most sensible devices: the NVIDIA A100 and T4 GPUs. A similar procedure would follow for the other processors.

We can approximate the $MTBF^{\text{CR}}$ caused by natural radiation (CR, cosmic ray induced). We use the σ obtained in experiments but also the natural neutron flux (F_{CR}) on the geographical location of the hardware. Then, the mean time between failures due to natural radiation reads:

$$MTBF^{\text{CR}} = \frac{1}{\sigma \times F_{\text{CR}}}. \quad (2)$$

Therefore, we also require calculating F_{CR} . It is important to mention again that the neutron interaction with the matter depends strongly on its energy, so the following estimation applies to the energy range provided by the neutron sources here used.

Machine	Top500 [J. 2022]	A100	location	alt. [m.a.s.l]	F_{CR} [cm ⁻² h ⁻¹]
Perlmutter	7	6,188	Berkeley, US	210	4.8
Selene	8	4,480	Sta. Clara, US	70	2.4
JUWELS B.	11	3,744	Jülich, DE	100	3.9
Polaris	14	2,240	Argonne, US	220	5.2
Chervonenkis	22	1,592	Moscow, RU	150	5.7
BioHive-1	106	320	Salt Lake C. US	1,300	13.4

Table 3 Location and calculated natural neutron flux (F_{CR}) for supercomputers relying on NVIDIA A100 GPUs and ranked in the Top500 list in June, 2022.

Bearing that in mind, first we obtained the natural neutron flux using the curve labelled as M of Fig 3.18 in Ref. [42]. We then fit the data to a decaying power law and integrated between 1 and 10 MeV to have an overlapping energy range with our sources (check Fig. 3). This led to a cosmic ray neutron flux $F_{CR} = 3.5 \text{ cm}^{-2} \text{ h}^{-1}$ at the sea level and 50° of latitude. Then, we calculated neutron flux for places where several supercomputers of interest are installed (see Table 3). For doing this, we calculated the expected flux at every location by correcting the corresponding latitude [43] and altitude [44] effects.

Fig. 8 shows the values of σ_{CE} for the NVIDIA A100. Taking into account that the σ is undervalued if Eq. 1 is used, we can round the averaged value to $\sigma_{CE} = 10^{-8} \text{ cm}^2$. As an application, knowing that the Perlmutter¹ supercomputer counts 6,188 A100 devices, we can expect at least a CE event every approximately every 4.6 months of operation. Other facilities with a lesser number of cards are more influenced by their placement. This is the case of the JUWELS Booster Module² and the Polaris³ supercomputer, which will suffer at least a CE crash every year. As you see in Table 3, the altitude is the most important factor (such as for BioHive-1⁴), but also the geomagnetic conditions in every location (Chervonenkis⁵).

The mean time between silent errors, $MTBF_{SE}^{CR}$, can also be estimated. From Table 1, we calculate the averaged σ_{SE} , resulting in $5.6 \times 10^{-9} \text{ cm}^2$ with a larger uncertainty estimation ($\sim 40\%$). Results for mean times between events are almost twice $MTBF_{CE}^{CR}$, as it can be seen in Table 4.

Similarly, we can also estimate the impact of CE on the NVIDIA T4 GPU. According to the values of Table 2 we also take $\sigma_{CE} = 10^{-8} \text{ cm}^2$ for the T4, and consequently, it will individually experiment, at least, a $MTBF_{CE}^{CR} \approx 3,200$ years at the sea level and 50° of latitude. However, the value lowers to ≈ 834 years in places such as Salt Lake City, US, where large companies devoted to AI business are established. Moreover, it is usual to extend the lifetime of a cluster replacing or installing improved hardware parts, such as SSD disks or more memory slots. T4 cards are cheaper and thinner than V100/A100 devices,

¹<https://www.nersc.gov/systems/perlmutter/>

²<https://www.fz-juelich.de/en/ias/jsc/systems/supercomputers/juwels>

³<https://www.alcf.anl.gov/polaris>

⁴<https://www.recursion.com/>

⁵<https://yandex.com/supercomputers>

Machine	$MTBF_{CE}^{CR}$ [days]	$MTBF_{SE}^{CR}$ [days]
Perlmutter	140	250
Selene	390	696
JUWELS B.	285	509
Polaris	360	643
Chervonenkis	459	819
BioHive-1	971	1734

Table 4 Estimated mean times between failures and silent errors of several Top500 supercomputers relying on NVIDIA A100 GPUs.

but a good choice to amortise past investments. Tetralith⁶ supercomputer at NSC in Sweden is an example of this. It was purchased in several phases in 2018 and it reached the Top500 74th position in June 2019, posteriorly it was upgraded with 170 T4 cards.

These numbers indicate that, unless a very big number of GPUs are working in parallel on the same problem, this kind of CE is very unlikely to seriously impact, with exception of large supercomputers. The estimation performed is larger but still comparable to the experiments documented in [36], although the hardware used in that work is older and more susceptible to CE. Regarding the catastrophic errors in CPUs, Table 1 indicates that the expected $MTBF_{CE}^{CR}$ is one order of magnitude larger than for GPUs. Additional work is required in characterizing the response of the device to neutrons energy < 1 MeV and > 10 MeV, which will reduce our estimation of $MTBF_{CE}^{CR}$ [2].

5 Conclusions

We have performed a series of irradiation experiments on a set of state-of-the-art processors commonly used for HPC (three GPU and two CPU models). Two neutron source, ²⁵²Cf and ²⁴¹Am-Be, are used for a total time of 190 hours of irradiation spanning 18 months. The hardware response to the radiation is characterised in terms of anomalies in their functioning. We have classified the errors as auto-corrected (AE), catastrophic (CE) and silent (SE), being the last class the most dangerous and undesirable. The experiment results are shown in Table 1.

Auto-corrected errors are harmless except for a possible small increase in the computation time, not detected in the current experiments. CE appeared mostly on GPUs, indicating that those devices are more sensitive to radiation than CPUs. On the other hand, the former ones are more prone to AE, as we can check in Figs. 6 and 8. The most sensitive GPU is the NVIDIA A100, followed by the T4 and finally the V100, which shows a higher resistance. Those errors are easily detected by the user or administrators since they simply stop the computation or the computer crashes. Checkpointing techniques are appropriate for dealing with such errors. The most positive result of this work

⁶<https://www.nsc.liu.se/systems/tetralith/>

is the lack of silent errors in all the computations completed, except maybe those in the NVIDIA A100 whose impact is still very low even in the only in presence of natural radiation (see Subsecs. 4.3 and 4.5 for discussion). We can also conclude that a more precise estimation of any *MTBF* or σ would require a larger number of AE/CE/SE events. This can be accomplished by increasing the irradiation time and/or using a more intense radioactive source.

Taking into account these results, the administrators of supercomputing facilities can improve the plans of contingencies needed to be implemented via software (or even from an architectonic point of view) for overcoming potential errors due to the natural neutron irradiation deposited in their infrastructure. Moreover, the researchers that are making use of these facilities can estimate the risk of silent errors for implementing resilient algorithms or checking their results with additional calculations, if were needed. A very approximate estimation of the mean time between failure for CE in part of the cosmic ray spectrum for the most sensible devices leads to thousands of years per unit. Thus, modern hardware combined with ECC techniques turns out to be very resilient to naturally occurring ionizing radiation at the sea level. However, the increased error rates at altitude and the weight excess due to shielding could make these hardware unfeasible for the aerospace field. Furthermore, as the *MTBF* is inversely proportional to the number of hardware elements, accidents would multiply in a World where the autonomous driving were predominant. For these reasons, shielding mechanisms and materials should be evaluated for those uses in future works.

According to the obtained experimental results, the ultimate hardware has incorporated automatic capabilities for overcoming silent errors, which were foreseen as critical a few years ago with the advent of exascale computing and the higher density of processors per square meter. In this sense, the lithography improvements are reducing the size of transistors, but our results suggest that size does not influence on the resistance of the device. Nowadays, vendors have greatly improved their product performance and silent errors would be scarcely produced, but they still occurs. Therefore, as new hardware continuously appears, it must be tested for radiation induced errors prior to its installation in large supercomputers.

Declarations

Ethical Approval

Not applicable.

Competing interests

The authors have no relevant financial, non-financial, proprietary or competing interests to declare that are relevant to the content of this article.

Authors' contributions

Andrés Bustos is the main contributor to the manuscript and experiments. Rafael Mayo-García, Antonio Juan Rubio-Montero, Roberto Méndez, and Hernán Asorey have reviewed the text, datasets, and physics, and they have

performed additional calculations. Experimental measurements in the Neutron Standards Laboratory were performed by its staff, Roberto Méndez, Sergio Rivera, Francisco González, and Xandra Campo.

Funding

This work was partially funded by the Spanish Ministry of Science and Innovation under contract number FIS2017-88892-P and RTI2018-096006-B-I00 (CODEC-OSE) with European Regional Development Fund (ERDF) funds, and by the Comunidad de Madrid CABAHLA-CM project (S2018/TCS-4423).

Availability of data and materials

Meaningful data generated or analysed during this study are included in this published article. Training datasets and the software used during this study are publicly available following the links included in the References section. Other raw datasets generated or analysed during the current study are available from the corresponding author on reasonable request.

References

- [1] Oliveira, D.A.G.D., Pilla, L.L., Hanzich, M., Fratin, V., Fernandes, F., Lunardi, C., Cela, J.M., Navaux, P.O.A., Carro, L., Rech, P.: Radiation-induced error criticality in modern HPC parallel accelerators. In: 2017 IEEE International Symposium on High Performance Computer Architecture (HPCA), pp. 577–588 (2017). <https://doi.org/10.1109/HPCA.2017.41>
- [2] Oliveira, D., Pilla, L., DeBardleben, N., Blanchard, S., Quinn, H., Koren, I., Navaux, P., Rech, P.: Experimental and analytical study of Xeon Phi reliability. In: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis. SC '17, pp. 1–12. Association for Computing Machinery, New York, NY, USA (2017). <https://doi.org/10.1145/3126908.3126960>
- [3] Mukherjee, S.S., Weaver, C., Emer, J., Reinhardt, S.K., Austin, T.: A systematic methodology to compute the architectural vulnerability factors for a high-performance microprocessor. In: Proceedings of the 36th Annual IEEE/ACM International Symposium on Microarchitecture. MICRO 36, pp. 29–40. IEEE Computer Society, USA (2003). <https://doi.org/10.1109/MICRO.2003.1253181>
- [4] Lu, G., Zheng, Z., Chien, A.A.: When is multi-version checkpointing needed? In: Proceedings of the 3rd Workshop on Fault-Tolerance for HPC at Extreme Scale. FTXS '13, pp. 49–56. Association for Computing Machinery, New York, NY, USA (2013). <https://doi.org/10.1145/2465813.2465821>

- [5] Lyons, R.E., Vanderkulk, W.: The use of triple-modular redundancy to improve computer reliability. *IBM Journal of Research and Development* **6**(2), 200–209 (1962). <https://doi.org/10.1147/rd.62.0200>
- [6] Moody, A., Bronevetsky, G., Mohror, K., Supinski, B.R.d.: Design, modeling, and evaluation of a scalable multi-level checkpointing system. In: *Proceedings of the 2010 ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis. SC '10*, pp. 1–11. IEEE Computer Society, USA (2010). <https://doi.org/10.1109/SC.2010.18>
- [7] 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 Seventeenth International Conference on Architectural Support for Programming Languages and Operating Systems. ASPLOS XVII*, pp. 111–122. Association for Computing Machinery, New York, NY, USA (2012). <https://doi.org/10.1145/2150976.2150989>
- [8] Bridges, P.G., Hoemmen, M., Ferreira, K.B., Heroux, M.A., Soltero, P., Brightwell, R.: Cooperative application/OS DRAM fault recovery. In: Alexander, M., D'Ambra, P., Belloum, A., Bosilca, G., Cannataro, M., Danelutto, M., Di Martino, B., Gerndt, M., Jeannot, E., Namyst, R., Roman, J., Scott, S.L., Traff, J.L., Vallée, G., Weidendorfer, J. (eds.) *Euro-Par 2011: Parallel Processing Workshops. LNCS*, vol. 7156, pp. 241–250. Springer, Berlin, Heidelberg (2012). https://doi.org/10.1007/978-3-642-29740-3_28
- [9] Bosilca, G., Delmas, R., Dongarra, J., Langou, J.: Algorithm-based fault tolerance applied to high performance computing. *Journal of Parallel and Distributed Computing* **69**(4), 410–416 (2009). <https://doi.org/10.1016/j.jpdc.2008.12.002>
- [10] Shantharam, M., Srinivasmurthy, S., Raghavan, P.: Fault tolerant preconditioned conjugate gradient for sparse linear system solution. In: *Proceedings of the 26th ACM International Conference on Supercomputing. ICS '12*, pp. 69–78. Association for Computing Machinery, New York, NY, USA (2012). <https://doi.org/10.1145/2304576.2304588>
- [11] Benson, A.R., Schmit, S., Schreiber, R.: Silent error detection in numerical time-stepping schemes. *The International Journal of High Performance Computing Applications* **29**(4), 403–421 (2015). <https://doi.org/10.1177/1094342014532297>
- [12] Fiala, D., Mueller, F., Engelmann, C., Riesen, R., Ferreira, K., Brightwell, R.: Detection and correction of silent data corruption for large-scale high-performance computing. In: *SC '12: Proceedings of the International*

- Conference on High Performance Computing, Networking, Storage and Analysis, pp. 1–12 (2012). <https://doi.org/10.1109/SC.2012.49>
- [13] Subasi, O., Di, S., Bautista-Gomez, L., Balaprakash, P., Unsal, O., Labarta, J., Cristal, A., Krishnamoorthy, S., Cappello, F.: Exploring the capabilities of support vector machines in detecting silent data corruptions. *Sustainable Computing: Informatics and Systems* **19**, 277–290 (2018). <https://doi.org/10.1016/j.suscom.2018.01.004>
 - [14] Wang, C., Dryden, N., Cappello, F., Snir, M.: Neural network based silent error detector. In: 2018 IEEE International Conference on Cluster Computing (CLUSTER), pp. 168–178. IEEE Computer Society, USA (2018). <https://doi.org/10.1109/CLUSTER.2018.00035>
 - [15] Chen, Z.: Online-ABFT: An online algorithm based fault tolerance scheme for soft error detection in iterative methods. *SIGPLAN Not.* **48**(8), 167–176 (2013). <https://doi.org/10.1145/2517327.2442533>
 - [16] Coleman, E., Jamal, A., Baboulin, M., Khabou, A., Sosonkina, M.: A comparison of soft-fault error models in the parallel preconditioned flexible GMRES. In: Wyrzykowski, R., Dongarra, J., Deelman, E., Karczewski, K. (eds.) *Parallel Processing and Applied Mathematics*. LNCS, vol. 10777, pp. 36–46. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-78024-5_4
 - [17] Sloan, J., Kumar, R., Bronevetsky, G.: An algorithmic approach to error localization and partial recomputation for low-overhead fault tolerance. In: 2013 43rd Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), pp. 1–12 (2013). <https://doi.org/10.1109/DSN.2013.6575309>
 - [18] Chakraborty, L., Singh, A.: A utility-based approach to cost-aware caching in heterogeneous storage systems. In: 2007 IEEE International Parallel and Distributed Processing Symposium, p. 50. IEEE Computer Society, Los Alamitos, CA, USA (2007). <https://doi.org/10.1109/IPDPS.2007.370240>
 - [19] Oliveira, D., dos Santos, F.F., Piscoya Dávila, G., Cazzaniga, C., Frost, C., Baumann, R.C., Rech, P.: High-energy versus thermal neutron contribution to processor and memory error rates. *IEEE Transactions on Nuclear Science* **67**(6), 1161–1168 (2020). <https://doi.org/10.1109/TNS.2020.2970535>
 - [20] Dueben, P.D., Joven, J., Lingamneni, A., McNamara, H., De Micheli, G., Palem, K.V., Palmer, T.N.: On the use of inexact, pruned hardware in atmospheric modelling. *Philosophical Transactions of the Royal Society A: Mathematical, Physical & Engineering Sciences* **372**(2018), 20130276

- (2014). <https://doi.org/10.1098/rsta.2013.0276>
- [21] Benoit, A., Cavelan, A., Cappello, F., Raghavan, P., Robert, Y., Sun, H.: Coping with silent and fail-stop errors at scale by combining replication and checkpointing. *Journal of Parallel and Distributed Computing* **122**, 209–225 (2018). <https://doi.org/10.1016/j.jpdc.2018.08.002>
- [22] Aupy, G., Benoit, A., Hérault, T., Robert, Y., Vivien, F., Zaidouni, D.: On the combination of silent error detection and checkpointing. In: *IEEE 19th Pacific Rim International Symposium on Dependable Computing, PRDC 2013, Vancouver, BC, Canada, December 2-4, 2013*, pp. 11–20. IEEE Computer Society, USA (2013). <https://doi.org/10.1109/PRDC.2013.10>
- [23] Benoit, A., Cavelan, A., Cappello, F., Raghavan, P., Robert, Y., Sun, H.: Identifying the right replication level to detect and correct silent errors at scale. In: *Proceedings of the 2017 Workshop on Fault-Tolerance for HPC at Extreme Scale. FTXS '17*, pp. 31–38. Association for Computing Machinery, New York, NY, USA (2017). <https://doi.org/10.1145/3086157.3086162>
- [24] Berrocal, E., Bautista-Gomez, L., Di, S., Lan, Z., Cappello, F.: Toward general software level silent data corruption detection for parallel applications. *IEEE Transactions on Parallel and Distributed Systems* **28**(12), 3642–3655 (2017). <https://doi.org/10.1109/TPDS.2017.2735971>
- [25] Aupy, G., Benoit, A., Cavelan, A., Fasi, M., Robert, Y., Sun, H., Uçar, B.: Coping with Silent Errors in HPC Applications. Adamatzky, A. (eds) *Emergent Computation. Emergence, Complexity and Computation*, vol. 24, pp. 269–292. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-46376-6_11
- [26] Ali, M.M., Strazdins, P.E., Harding, B., Hegland, M.: Complex scientific applications made fault-tolerant with the sparse grid combination technique. *The International Journal of High Performance Computing Applications* **30**(3), 335–359 (2016). <https://doi.org/10.1177/1094342015628056>
- [27] Toueg, S., Babaoglu, O.: On the optimum checkpoint selection problem. *SIAM Journal on Computing* **13**, 630–649 (1984). <https://doi.org/10.1137/0213039>
- [28] Losada, N., González, P., Martín, M.J., Bosilca, G., Bouteiller, A., Teranishi, K.: Fault tolerance of MPI applications in exascale systems: The ULFM solution. *Future Generation Computer Systems* **106**, 467–481 (2020). <https://doi.org/10.1016/j.future.2020.01.026>

- [29] Fasi, M., Langou, J., Robert, Y., Uçar, B.: A Backward/Forward Recovery Approach for the Preconditioned Conjugate Gradient Algorithm. Research Report RR-8826, ENS Lyon, CNRS & INRIA (December 2015). <https://hal.inria.fr/hal-01242327>
- [30] Yao, E., Zhang, J., Chen, M., Tan, G., Sun, N.: Detection of soft errors in LU decomposition with partial pivoting using algorithm-based fault tolerance. *The International Journal of High Performance Computing Applications* **29**(4), 422–436 (2015). <https://doi.org/10.1177/1094342015578487>
- [31] Canal, R., Hernandez, C., Tornero, R., Cilaro, A., Massari, G., Reghenzani, F., Fornaciari, W., Zapater, M., Atienza, D., Oleksiak, A., Piundefedtek, W., Abella, J.: Predictive reliability and fault management in exascale systems: State of the art and perspectives. *ACM Comput. Surv.* **53**(5) (2020). <https://doi.org/10.1145/3403956>
- [32] Heldens, S., Hijma, P., Werkhoven, B.V., Maassen, J., Belloum, A.S.Z., Van Nieuwpoort, R.V.: The landscape of exascale research: A data-driven literature analysis. *ACM Comput. Surv.* **53**(2) (2020). <https://doi.org/10.1145/3372390>
- [33] Ziegler, J.F., Nelson, M.E., Shell, J.D., Peterson, R.J., Gelderloos, C.J., Muhlfeld, H.P., Montrose, C.J.: Cosmic ray soft error rates of 16-MB DRAM memory chips. *IEEE Journal of Solid-State Circuits* **33**(2), 246–252 (1998). <https://doi.org/10.1109/4.658626>
- [34] Baumann, R.C.: Radiation-induced soft errors in advanced semiconductor technologies. *IEEE Transactions on Device and Materials Reliability* **5**(3), 305–316 (2005). <https://doi.org/10.1109/TDMR.2005.853449>
- [35] Vázquez-Luque, A., Marin, J., Terrón, J., Cameán, M., Bedogni, R., Sánchez-Doblado, F., Gómez, F.: Neutron induced single event upset dependence on bias voltage for CMOS SRAM with BPSG. *Nuclear Science, IEEE Transactions on* **60**, 4692–4696 (2013). <https://doi.org/10.1109/TNS.2013.2283532>
- [36] Tiwari, D., Gupta, S., Rogers, J., Maxwell, D., Rech, P., Vazhkudai, S., Oliveira, D., Londo, D., DeBardeleben, N., Navaux, P., Carro, L., Bland, A.: 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), pp. 331–342 (2015). <https://doi.org/10.1109/HPCA.2015.7056044>
- [37] Méndez-Villafañe, R., Guerrero, J.E., Embid, M., Fernández, R., Grandio, R., Pérez-Cejuela, P., Márquez, J.L., Alvarez, F., Ortego, P.: Design and verification of the shielding around the new neutron standards laboratory

- (LPN) at CIEMAT. Radiation protection dosimetry **161**(1-4), 393–397 (2014). <https://doi.org/10.1093/rpd/nct348>
- [38] The International Organization for Standardization: ISO 8529-1 Neutron reference radiations fields - Part 1: Characteristics and methods of production (2021). <https://www.iso.org/standard/80060.html>
- [39] von Mises, R., Pollaczek-Geiringer, H.: Praktische verfahren der gleichungsauflösung . Zamm-zeitschrift Fur Angewandte Mathematik Und Mechanik **9**, 58–77 (1929). <https://doi.org/10.1002/zamm.19290090105>
- [40] Xiao, H., Rasul, K., Vollgraf, R.: Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms. Comment: Dataset is freely available at <https://github.com/zalandoresearch/fashion-mnist> and Benchmark is available at <http://fashion-mnist.s3-website.eu-central-1.amazonaws.com/>. arxiv:1708.07747 (2017). <http://arxiv.org/abs/1708.07747>
- [41] Asorey, H., Mayo-García, R.: Calculation of the high energy neutron flux for anticipating errors and recovery techniques in exascale supercomputer centres. The Journal of Supercomputing (2022). <https://doi.org/10.1007/s11227-022-04981-8>
- [42] Grieder, P.K.: Cosmic Rays at Earth. Elsevier Science, Amsterdam (2001). <https://doi.org/10.1016/B978-0-444-50710-5.X5000-3>
- [43] Rojdev, K., Koontz, S., Reddell, B., Atwell, W., Boeder, P.: Comparison and validation of FLUKA and HZETRN as tools for investigating the secondary neutron production in large space vehicles. In: AIAA SPACE 2015 Conference and Exposition, p. 4601 (2015). <https://doi.org/10.2514/6.2015-4601>
- [44] Dilillo, L., Wrobel, F., Galliere, J.-M., Saigné, F.: Neutron detection through an sram-based test bench. In: 2009 3rd International Workshop on Advances in Sensors and Interfaces, pp. 64–69 (2009). <https://doi.org/10.1109/IWASI.2009.5184769>. IEEE