Evaluating the Impact of Mixed-Precision on Fault Propagation for Deep Neural Networks on GPUs

Fernando Fernandes dos Santos, Paolo Rech, Angeliki Kritikakou, Olivier Sentieys

Evaluating the Impact of Mixed-Precision on Fault Propagation for Deep Neural Networks on GPUs

Fernando Fernandes dos Santos[1], Paolo Rech[2], Angeliki Kritikakou[1], and Olivier Sentieys[1]
[1]Univ. Rennes, Inria, France. [2]University of Trento, Italy.

Graphics Processing Units (GPUs) offer the possibility to execute floating-point operations (FLOP) with **mixed-precisions** such as INT8, FP16, Bfloat, FP32, and FP64. For Deep Neural Networks (DNNs), a reduced precision is likely to lower the execution time and power consumption as it requires a smaller hardware area and fewer clock cycles to perform instructions than the standard FP32 and FP64 precisions. As less area is needed for reduced precision, the circuit error rate is also expected to be lower [1]. NVIDIA GPUs also have tensor cores that perform matrix multiplication on hardware. The tensor cores are capable to perform a 4×4 FP16 matrix multiplication in *one* clock cycle [2]. The tensor cores can deliver up to $9\times$ higher performance than the software implementation of matrix multiplication (sequence of sums and multiplications) on GPUs and up to $47\times$ than a CPU-based system [2].

However, the impact of a fault in reduced-precision data could be much more severe than corruption in full-precision data [3]. As DNNs are used to detect and classify objects in safety-critical systems, their reliability needs to be carefully evaluated [4]. Furthermore, like any other electronic device, modern GPUs are susceptible to transient faults induced by neutrons [5], [6]. The impact of neutrons on the hardware can change the transistor state leading to bit flips in the memories or spikes on logic circuits [7]. The fault can lead to: (1) Silent Data Corruption (SDC), where the application generates an incorrect output without a flag or indication of error, (2) system operation interruption, crashes, and application hangs, or (3) no visible effect on the system, that is, the fault is masked. Researchers expose the device to a beam of neutrons to evaluate the error rate of the codes running on GPUs [6]. The accelerated particle beam induces transient faults in the device hardware. As the whole chip irradiated beam experiments provide the realistic error rate of the device running a code.

To measure the error rate of mixed precision algorithms related to DNNs, we have exposed an NVIDIA Volta GPU (Tesla V100) to a beam of neutrons on the ChipIR facility at the Rutherford Appleton Laboratory (RAL) in Didcot, UK. We chose multiple floating-point precisions of a General Matrix Multiplication (GEMM), such as FP16, FP32, FP64, and tensor cores using FP16. The choice of multiple GEMM configurations is guided as they are the core of the state-of-the-art DNNs. Additionally, as a study case, we also exposed an object detection DNN, YOLOv3, with two precisions, FP16 and FP32.

Funding: The European Union's Horizon 2020 research and programme under the MSCA grant agreement No 899546. And CAPES, Brazil.

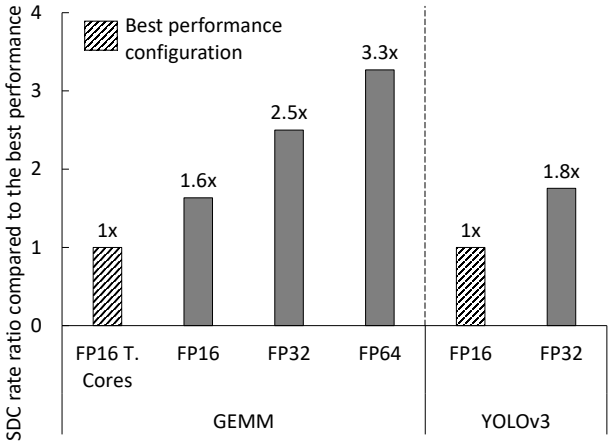

Fig. 1. SDC rate ratio between the best performance GEMM and YOLOv3, compared to larger precisions on Tesla V100.

Figure 1 shows the ratio of the SDC rate increase compared to the best performance configuration. That is, GEMM with FP16 executing on Tensor cores has the lowest execution time on all GEMMs configurations. Thus the error rate ratio is calculated using GEMM FP16 Tensor Cores as the baseline. The same is presented for YOLOv3 with FP16.

Not surprisingly, the smaller the precision, the smaller the execution time and error rate. As the lower precisions use less resources, achieving higher performance and lower error rates is possible. Additionally, the tensor cores with an FP16 precision configuration leads to a lower error rate than the software implementation of GEMM FP16. The better usage of the resources can improve performance and the error rate.

References

[1] F. Fernandes dos Santos *et al.*, "Reliability evaluation of mixed-precision architectures," in *IEEE HPCA*, 2019.
[2] Z. Jia *et al.*, "Dissecting the NVIDIA volta GPU architecture via microbenchmarking," 2018.
[3] F. Libano *et al.*, "How reduced data precision and degree of parallelism impact the reliability of convolutional neural networks on fpgas," *IEEE Transaction on Nuclear Science*, 2021.
[4] Y. Ibrahim *et al.*, "Soft error resilience of deep residual networks for object recognition," *IEEE Access*, 2020.
[5] J. Tan *et al.*, "Analyzing soft-error vulnerability on gpgpu microarchitecture," in *IEEE IISWC*, 2011.
[6] P. Rech *et al.*, "Impact of gpus parallelism management on safety-critical and hpc applications reliability," in *IEEE/IFIP DSN*, 2014.
[7] R. Baumann, "Soft errors in advanced computer systems," *IEEE Design Test of Computers*, 2005.