IL NUOVO CIMENTO **47 C** (2024) 133
DOI 10.1393/ncc/i2024-24133-9

Colloquia: IFAE 2023

FPGA inference of Deep Neural Network-based trigger algorithms at Colliders(*)

L. Rambelli(1)(3)(**), A. Coccaro(1), F. Di Bello(1)(3), S. Giagu(2) and N. Stocchetti(2)

(1) *INFN, Sezione di Genova - Genova, Italy*
(2) *Dipartimento di Fisica, Università di Roma "La Sapienza" - Roma, Italy*
(3) *Dipartimento di Fisica, Universitá di Genova - Genova, Italy*

received 13 February 2024

Summary. — Experimental particle physics demands a sophisticated trigger and acquisition system capable to efficiently retain the collisions of interest for further investigation. Heterogeneous computing with the employment of FPGA cards may emerge as a trending technology for the triggering strategy of the upcoming high-luminosity program of the Large Hadron Collider at CERN. In this context, this work presents two machine-learning algorithms for selecting events where neutral long-lived particles decay within the detector volume studying their accuracy and inference time when accelerated on commercially available Xilinx FPGA accelerator cards. The inference time is also compared with a CPU- and GPU-based hardware setup. The results indicate that all tested architectures fit within the accuracy and latency requirements of a second-level trigger farm and that exploiting accelerator technologies for real-time processing of particle-physics collisions is a promising research field that deserves additional investigations, in particular with machine-learning models with a large number of trainable parameters.

1. – Introduction

The trigger and data acquisition system is a crucial aspect in experimental particle physics at colliders. It is challenging to efficiently collect collision data for analysis due to the complexity of the detector data and the need for quick processing. The ATLAS and CMS experiments at CERN's Large Hadron Collider (LHC) use a two-tier trigger system to select collision events for storage and analysis. The initial 40 MHz proton-proton collision rate produced by the LHC is first reduced to around 100 kHz by a hardware-based Level-1 (L1) trigger system, and further reduced to around 1 kHz by a software High Level Trigger (HLT), optimizing event selection while considering latency, throughput, data transfer, and storage capabilities. With the upcoming high-luminosity LHC (HL-LHC) phase, new design solutions, such as FPGA-accelerated machine learning

(*) IFAE 2023 - "Poster" session
(**) Speaker.

inference, may be needed to handle the increased occupancy and readout channels of the upgraded detectors.

In this context, this work studies the possibility to implement Deep Neural Network (DNN) based algorithms for the event selection at the HLT, and to use commercial accelerator boards based on FPGA processors to improve the performance in terms of processing time and throughput. FPGAs are reconfigurable hardware architectures which can be adapted for specific tasks and are traditionally programmed using hardware description languages like VHDL or Verilog. In recent years several tools and libraries were developed to facilitate the implementation and deployment of both traditional and machine learning algorithms on FPGAs, like the Vitis-AI tool released by the Xilinx company that is used in this work.

This work shows different DNN-based models for targeting the selection of events where neutral long-lived particles decay within the detector volume. We present the design and the results of the implementation in a working engineering pipeline that starts from the pre-processing of the input data, to the training of the DNN-based model, to the optimization and deployment on two Xilinx FPGA accelerators, the Alveo U50 and the Alveo U250, all based on the use of publicly available libraries. Two approaches based on a deep convolutional neural network and on an autoencoder are developed and presented. A comparison of the performances of the deployed algorithms in CPU, GPU and FPGA accelerators is also shown. This work is also available on the Machine Learning: Science and Technology journal as a paper titled "Fast Neural Network Inference on FPGAs for Triggering on Long-Lived Particles at Colliders" [1].

2. – Physics benchmark and datasets

This work focuses on the identification of neutral long-lived particles (LLPs), arising from a variety of beyond the SM scenarios proposed in literature, with the data collected by the muon spectrometer (MS) of a typical experiment at the LHC. A toy simulation of the monitored drift tube (MDT) detector together with the superconducting toroidal magnetic field of the ATLAS experiment is developed, together with the physics benchmark of a neutral LLP decaying to charged particles.

Physics processes are simulated with a number of charged particles as decay products from two to ten, representative of the cases of two-body and multi-body decays of a X particle with a uniformly distributed decay length L_r in the range [0, 5] m.

Images has vertical bin size equal to 20, that corresponds to the number of MDT chambers layers, while the horizontal axis is set to 333 and represents a realistic average number of MDT tubes in the ATLAS detector. For each choice of charged particle multiplicity, 5k images are generated separately, with a total of 45k available events. The sample is randomly split in two parts so that 80% of the images are employed for models training and the remaining 20% for the evaluations.

3. – Neural network models

In this work two algorithms representative of two different triggering philosophies are developed and characterised. A deep convolutional neural network (CNN) is trained for regressing the L_r parameter of the neutral LLP while an autoencoder (AE) is trained exclusively on events where the decays of the LLP occurred near the interaction point for detecting anomalies. Once trained and deployed in the trigger and acquisition system, the CNN and the AE can be employed to define a selection criteria based, in the first

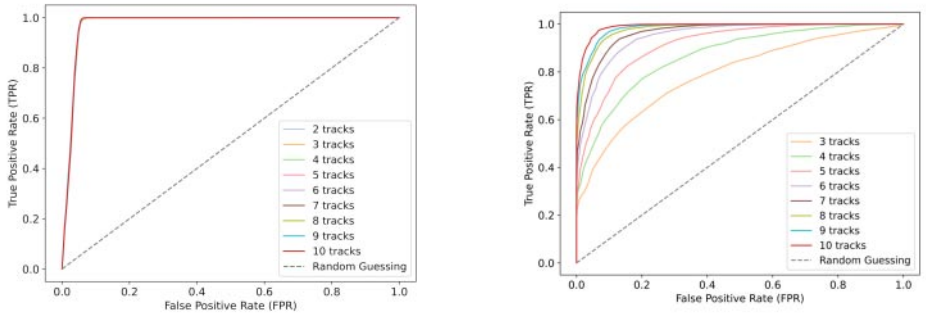

Fig. 1. – ROC curves for the CNN (left) and the AE (right) models. LLP decays are labelled as signal if $3\,\mathrm{m} < L_r < 5\,\mathrm{m}$ and as background if $0\,\mathrm{m} < L_r < 1\,\mathrm{m}$. Track multiplicity between two and ten, and between two and four, is used for creating the background dataset, respectively for the CNN and the AE models. In contrast, the track multiplicity is considered separately for signal.

case, on the inferred L_r parameter and, in the second case, on the likelihood of the event to not only contain prompt decays.

The CNN model comes with ∼2.8M trainable parameters, while the AE model with ∼398k, and ∼162k for the encoder part. We highlight how the chosen architectures are not ideal for the typical sparsity and cardinality of the data emerging from particle-physics collisions; they were chosen, instead, because they are fully supported by the adopted publicly available libraries.

4. – Results

Models performance were studied in terms of prediction accuracy, signal efficiency and background rejection, and the Receiver Operating Curves (ROCs) were produced for both models for different number of tracks in the final state.

For inference time comparison, two FPGA boards are considered: Xilinx Alveo U50 and U250. The AE model couldn't be run on the U250 due to lack of support for the Reshape layer in Xilinx Vitis-AI tool. It's important to note that the server configurations for the two cards are different, making direct performance comparison between U50 and U250 for the CNN model difficult. To accelerate the FPGA cards, preliminary operations are required. The Xilinx distributed Vitis-AI tool provides a complete workflow for this purpose, by the quantization step to the final inference one.

The ROC curves demonstrate the capability of the CNN model to effectively learn the decay position independently of the multiplicity of the charged decay products, while a dependence on the multiplicity is clearly evident for the AE model.

The inference time and the throughput of the CNN and AE models on different architectures are also studied and results are presented in table I and were achieved using a consistent batch size value for CPU and GPU based deployment and for them the models were converted into the Open Neural Network Exchange (ONNX) format with the runtime engines corresponding to these two architectures. The measurements on the CPU were performed using all the cores and on a machine equipped with AMD EPYC 7302 16-Core processors. The measurements on the GPU were performed on a

TABLE I. – *Models inference time in ms and throughput in frames per second on different target architectures.*

CNN Model	CPU	GPU	U50	U250
Inference time [ms]	5.1 ± 1.1	1.0 ± 0.1	3.7 ± 0.1	3.1 ± 0.4
Throughput [fps]	302 ± 4	9930 ± 187	950 ± 5	553 ± 4
AE Model	CPU	GPU	U50	U250
Inference time [ms]	0.7 ± 0.1	0.41 ± 0.01	2.6 ± 0.3	/
Throughput [fps]	3477 ± 210	79238 ± 2358	1497 ± 3	/

GPU NVIDIA Tesla V100, and using the float models before quantization. The inference time results are obtained by averaging on few tens of measurements. The throughput is estimated by inferring the models with 10k images. The first measurements of both inference time and throughput on accelerators are discarded since they were observed to be systematically higher.

5. – Conclusions

Overall the study indicates that all architecture technologies offer inference time and throughput adequate for the typical latency requirements of a high-level trigger selection in a general-purpose experiment at LHC or HL-LHC. The inference time for the CNN model suggests that the acceleration on FPGA gives an advantage compared to the CPU-based approach. A similar advantage is not evident for the AE model. This can be attributed to the lightness of the model in terms of number of parameters, which results in the actual inference time being negligible compared to the time needed for loading the data onto the FPGA itself.

The throughput measurements also indicate the superiority of the FPGA-acceleration approach compared to the CPU-based one for the CNN model, and not for the AE model for the same considerations just expressed. In addition the throughput on the GPU architecture seems to suggest the superiority of this approach but this is achieved, as the corresponding measurements on the inference time confirm, only thanks to the capability of GPUs to process inference concurrently, and such high degree of concurrent computing can't be directly injected within a multi-node high-level trigger farm at colliders. All results are available also in the published paper in ref. [1].

REFERENCES

[1] COCCARO A. *et al.*, *Mach. Learn.: Sci. Technol.*, **4** (2023) 045040, https://arxiv.org/abs/2307.05152.