GNNs on FPGAs for Real-time Particle Reconstruction at the ATLAS Hardware Trigger

J. Lerner[1,2], M. Swiatlowski[1], and W. Fedorko[1]

[1]TRIUMF, Vancouver, BC, Canada, [2]University of British Columbia, Vancouver, BC, Canada

ATLAS EXPERIMENT

Abstract

With over 100 million proton collisions at the LHC every 25 ns, it is essential to be selective with the data that is kept for analysis. The ATLAS Hardware Trigger must reduce 40 Mhz of data down to just 1 MHz, saving only the most interesting events– the rest, lost forever. New developments in the use of field programmable gate arrays (FPGAs) for hardware triggers have welcomed in opportunities to improve event selection with more complex machine learning algorithms, including GNNs. Demonstrated here is an initial exploration of the capabilities and limitations of a GarNet model built for Vivado using `hls4lm`.

Model Architecture

The GarNet [2] algorithm is designed to learn the unique detector geometries and take advantage of its sparse structure for fast and compact reconstruction.

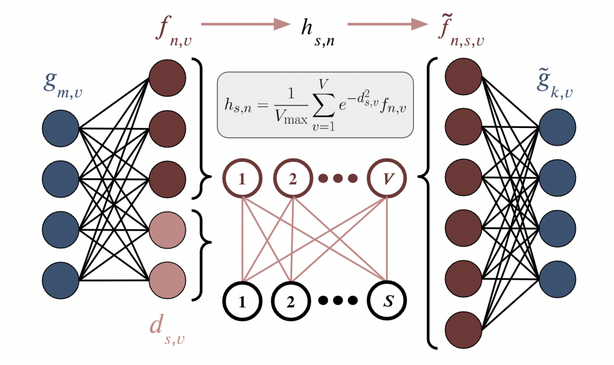

Figure 1: Data flow of the GarNet algorithm

1. The input features, $g_{m,v}$, are encoded as learned features, $f_{n,v}$, and distance parameters, $d_{s,v}$, using linear transformations.
2. A complete bipartite graph is built from the set of V vertices and S aggregators with edge weights, $W_{s,v} = e^{-d_{s,v}^2}$.
3. The learned features are averaged at each aggregator, $h_{s,n}$, and passed back as aggregated features, $\tilde{f}_{s,n,v} = W_{s,v} h_{s,n}$.
4. Aggregated features are decoded as output features, $\tilde{g}_{k,v}$.

Data Generation

- Over 15 million isolated clusters from charged and neutral single pion events
- Filtered to exclude cluster energies below 0.5 GeV and negative energy cells.
- Processed inputs: cluster energy, cell count, and up to $V_{\max}$ cells $(\eta,\ \phi,\ s,\ E)$

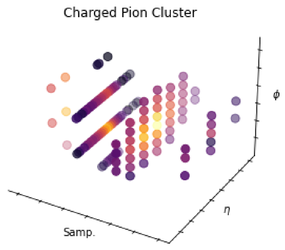

Figure 2: A graphical representation of a charged pion cluster with 128 cells

Training and Testing

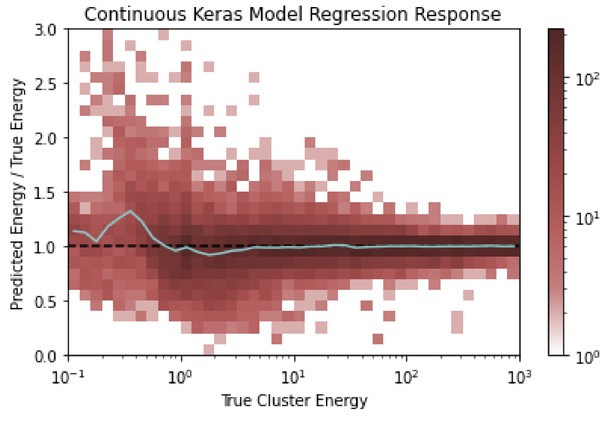

Figure 3: Energy reconstruction by 128 cell continuously trained GarNet model

- 99% mean squared error and 1% binary crossentropy weighted loss
- Cluster energy and 3 GarNet layers feed into dense extraction layers
- $V_{\max}$ cutoffs set to 32, 64, and 128 cells for inferences about scalability
- Quantization aware training with `QKeras` to reduce discretization issues

High Level Synthesis

- `hls4ml` [1] allows for fine control over the fixed point precision at every layer

$$\text{S} < \text{width bits, integer bits} >: \underbrace{001101}_{\text{integer}}.\underbrace{1010000000}_{\text{fractional}} = 13.625$$

- Converts `Keras/QKeras` models to firmware for implementation on FPGAs
- Compares accuracy, latency, and resources across models and precisions

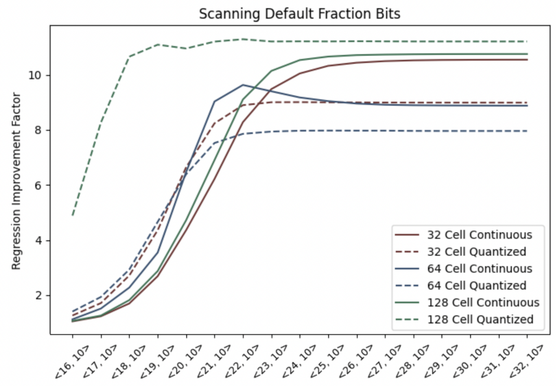

Figure 4: Performance of several models at different fixed precisions

Discussion

A promising workflow to bring the predictive power of the GarNet model to high-speed firmware has been shown, yielding greater than ten-fold increases in regression accuracy at less than half the bit-width and FPGA-deployable models with comparable performance to the continuous `Keras` model. While latency and resource consumption are expected to follow the opposite trend to performance with regards to precision, maximum cell count holds the tightest limit on available parallel processes, directly impacting latency. There are several further optimizations to be explored, including prunable parameters, collapsing layers, and the many hyperparameters in the GarNet algorithm.

References

[1] J. Duarte et al. Fast inference of deep neural networks in FPGAs for particle physics. *JINST*, 13(07), 2018.

[2] Y. Iiyama et al. Distance-Weighted Graph Neural Networks on FPGAs for Real-Time Particle Reconstruction in High Energy Physics. *Frontiers in Big Data*, 3, Jan 2021.