IRIS-HEP Fellowship Proposal: Developing a Python Front-End for HLS Implementation of GNNs on FPGA

Abdelrahman Elabd , University of Pennsylvania
Duration: January 4, 2021 - July 3, 2021
Area: Innovative Algorithms

1 Proposal

Graph Neural Networks (GNNs) have become an increasingly popular solution for particle tracking due to their efficiency at encoding context within physical systems. However, as the LHC is upgraded to achieve higher luminosity, faster and cheaper algorithms are required to handle the increasing data rates. One solution is to implement these GNN models on FPGAs, and ongoing efforts to do this include the Exa.TrkX project and the Accelerated GNN Tracking project at IRIS-HEP. The pace of research in machine learning calls for continuous implementation and testing of new GNN models and training paradigms, but this process is bottlenecked by the rate at which researchers can design the relevant HLS implementations. A quicker, more-abstracted tool is required.

I propose to develop a Python-based front-end for HLS implementation of GNN models on FPGA and integrate it into the `hls4ml` toolkit. Vesal Razavimaleki (UCSD) has already been working on HLS implementations for some of the GNNs used at IRIS-HEP. This project proposes to manually convert these existing HLS designs into pytorch.geometric models and use them as a benchmark to develop a python front-end that takes a trained pytorch.geometric model as its input, constructs the relevant IP blocks, and outputs an identical HLS implementation.

An important deliverable of this project will be a pull request against the official `hls4ml` project base that integrates the front-end described here toward a next release of `hls4ml` with this functionality.

The proposed work will be mentored by Mark Atkinson (University of Illinois at Urbana-Champaign) with additional collaborators within the IRIS-HEP GNN effort. In addition to Markus, I will work closely with Mark Neubauer (UIUC), Javier Duarte (UCSD) and Vesal Razavimaleki (UCSD) on this project.

2 Timeline

- **Week 1–2**: Review the HLS GNN models implemented by Vesal.
- **Week 3–4**: Use these HLS models to define 1 or 2 baseline pytorch.geometric models.
- **Week 5–10**: Implement within `hls4ml` a python-based front-end to translate the above pytorch.geometric models into the HLS template format
- **Week 11-12**: Help to prepare a pull request to the central `hls4ml` repository for above conversion front-end python code.

- **Week 13-18**: Use Brevitas[1] to create quantization-aware trained versions of the baseline pytorch.geometric models. Compare different paradigms of heterogeneous quantization-aware training [2] (i.e. a different quantization level for each layer). Use this to optimize baseline models.
- **Week 19-20**: Present work in Fast ML and IRIS-HEP Meetings.

3 References

1. https://github.com/Xilinx/brevitas

2. Claudionor N. Coelho Jr, Aki Kuusela, Hao Zhuang, Thea Aarrestad, Vladimir Loncar, Jennifer Ngadiuba, Maurizio Pierini, and Sioni Summers. 2020. CERN: Ultra Low-latency, Low-area Inference Accelerators using Heterogeneous Deep Quantization with QKeras and hls4ml. arXiv:2006.10159 https://arxiv.org/abs/2006.10159