IRIS-HEP Fellowship Proposal: Adapting GNN Tracking for FPGAs with `hls4ml`

Vesal Razavimaleki, UC San Diego

Duration: July 6, 2020 - September 25, 2020
Area: Innovative Algorithms

1 Proposal

Graph neural networks (GNNs) have demonstrated promise in pattern recognition problems such as particle tracking. One effective, but computationally expensive, example is the Exa.TrkX GNN segment classifier model [1], implemented in the `graph_nets` library [2] and `TensorFlow`. This model accepts an input graph composed of hits in a generic tracking detector. The node features are the cylindrical coordinates (r, ϕ, z) of the hits. The edges connect hits on adjacent layers of the detector (satisfying geometrical constraints) with differences of coordinates ($\Delta\eta$, $\Delta\phi$) as the edge features. The edges are labeled with a 1 if the two hits are part of the same track and 0 if they are not. The target of the GNN model is to correctly classify these segments.

To meet the demands and greater pileup of the planned HL-LHC, there has been interest in integrating machine learning (ML) methods into the L1 trigger and to accelerate large ML models with FPGA coprocessors. The deployment of neural networks in FPGAs has been studied with the `hls4ml` compiler package, which uses high-level synthesis (HLS) to convert ML models into FPGA firmware [3]. Preliminary investigation of GNNs in `hls4ml` has begun with GarNet [4], a lightweight GNN for clustering and calorimetry.

I propose to expand the `hls4ml` toolkit to support the Exa.TrkX GNN and similar architectures for particle tracking. To do so, the Exa.TrkX model must be fine-tuned and compressed such that the algorithm fits within realistic FPGA resources. This compression will facilitate implementation in HLS and integration into `hls4ml`. Under the local supervision and guidance of Javier Duarte (UC San Diego) with additional mentors and collaborators within IRIS-HEP Isobel Ojalvo (Princeton), Savannah Thais (Princeton), and Mark Neubauer (UIUC), I will prototype and develop a simplified HLS implementation of the Exa.TrkX segment classifier model, including generic GNN layers like "InteractionNetwork," which can support conversion from other GNN libraries. An important deliverable of my work will be a new release of `hls4ml` featuring these GNN layers, which can be used for L1 trigger or FPGA coprocessing applications. As the GNN model is very large, we will first focus on a minimal, small implementation of the algorithm and then scale it up to better understand the limits of FPGA-based GNN processing.

2 Timeline

- **Week 1–2**: Fork `hls4ml` repository and add HLS project for simplified Exa.TrkX model. Implement edge connections as lists of sender and receiver nodes. Test different pipelining options.
- **Week 3–5**: Explore further simplifications and compression of segment classifier model such as merging decoding and output networks. Research methods of compressing GNN with binary/ternary neural networks and trained quantization [5].
- **Week 6–8**: Finalize HLS implementation. Include "InteractionNetwork" and "GraphIndependent" `hls4ml` layers for conversion of similar models from other libraries.

- **Week 9–12**: Test HLS implementation with TrackML data set on FPGA. Prepare a pull request to add GNN layers to `hls4ml` repository, including example Exa.TrkX model, unit tests, and documentation. Present work in Fast ML and IRIS-HEP Meetings.

References

[1] X. Ju et al., "Graph neural networks for particle reconstruction in high energy physics detectors", in *Machine Learning and the Physical Sciences Workshop at the 33rd Annual Conference on Neural Information Processing Systems*. 2019. `arXiv:2003.11603`.

[2] P. W. Battaglia et al., "Relational inductive biases, deep learning, and graph networks", `arXiv:1806.01261`.

[3] J. Duarte et al., "Fast inference of deep neural networks in FPGAs for particle physics", *J. Instrum.* **13** (2018) P07027, `doi:10.1088/1748-0221/13/07/P07027`, `arXiv:1804.06913`.

[4] S. R. Qasim, J. Kieseler, Y. Iiyama, and M. Pierini, "Learning representations of irregular particle-detector geometry with distance-weighted graph networks", *Eur. Phys. J. C* **79** (2019) 608, `doi:10.1140/epjc/s10052-019-7113-9`, `arXiv:1902.07987`.

[5] G. Di Guglielmo et al., "Compressing deep neural networks on FPGAs to binary and ternary precision with `hls4ml`", `doi:10.1088/2632-2153/aba042`, `arXiv:2003.06308`. Accepted by *Mach. Learn: Sci. Tech.*