Neubauer Group Overview

Dewen Zhong (UIUC)

NSF Site Visit
July 11-12 2023
https://indico.cern.ch/event/1282754

OAC-2117997

https://a3d3.ai/

Our Group

UNIVERSITY OF ILLINOIS URBANA-CHAMPAIGN

NCSA

a3d3 Accelerated AI Algorithms for Data-Driven Discovery

ATLAS EXPERIMENT

Mark Neubauer
(Prof)

Markus Atkinson (Postdoc)

GNN-based EF tracking

Dewen Zhong* (PhD student)

Analog AI, ML Boosted WW tagging, HH/SH searches in bbWW decay channel

Casey Smith
(ECE, Engineer)

GNN-based EF tracking on FPGA, VHDL design

Avik Roy (Postdoc)

Explainable AI, FAIR, Anomaly-aware ML for trigger, Vector-like quark searches

Jared Burleson* (PhD student)

GNN-based EF tracking, Vector Boson Scattering, Boson Polarization-aware ML

Ben Galewsky
(NCSA Software Engineer)

Caching, Columnar Data Delivery (ServiceX)

Santosh Parajuli* (Postdoc)

GNN-based EF tracking

Jiangcong Zeng (PhD student)

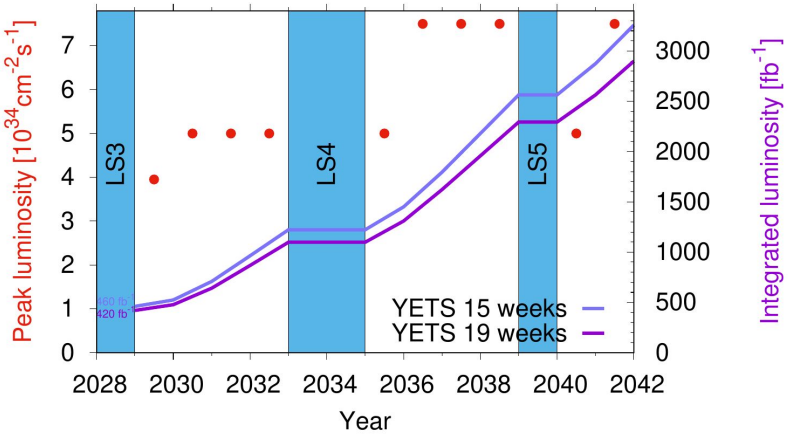
Vector Boson Scattering in semileptonic decay channel

* a3d3 - supported

HL-LHC Computing challenge

Accelerated AI Algorithms for Data-Driven Discovery

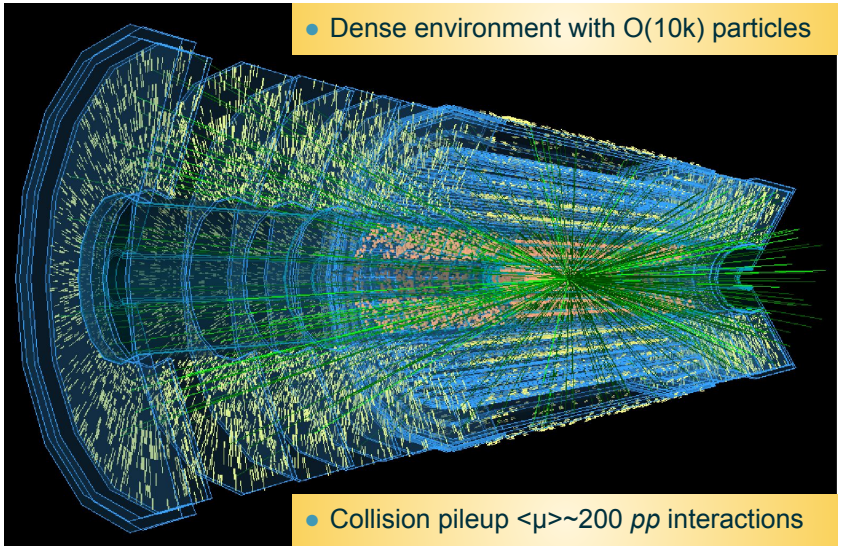
Efficient computational strategies are paramount for devices in resource-limited settings, particularly within high-energy physics experiments.
During the HL-LHC era, 10x more data per second than Run 1 & 2.

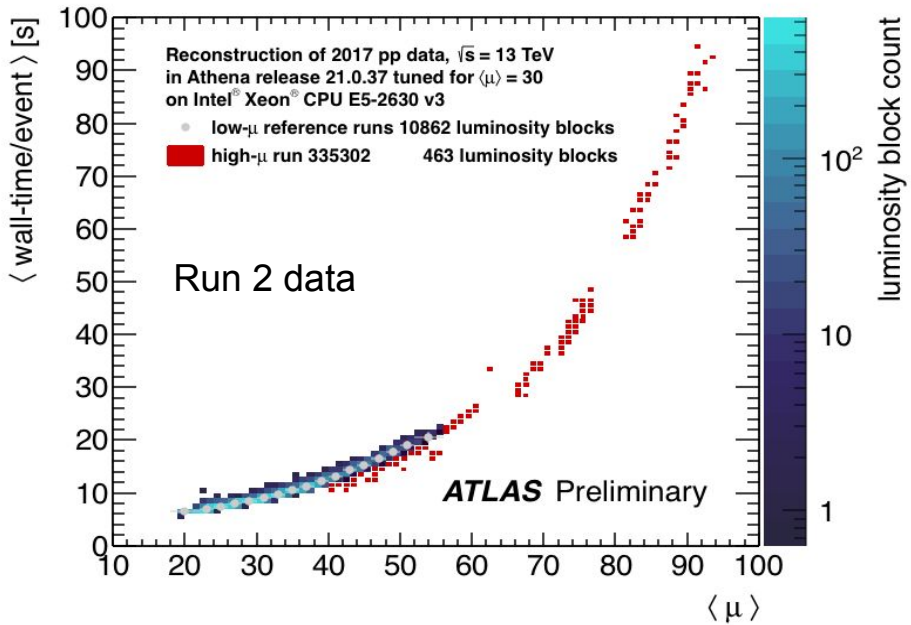


HL-LHC proton-proton luminosity out to 2041

Tracking challenge

Simulated pp collision event in ATLAS during HL-LHC



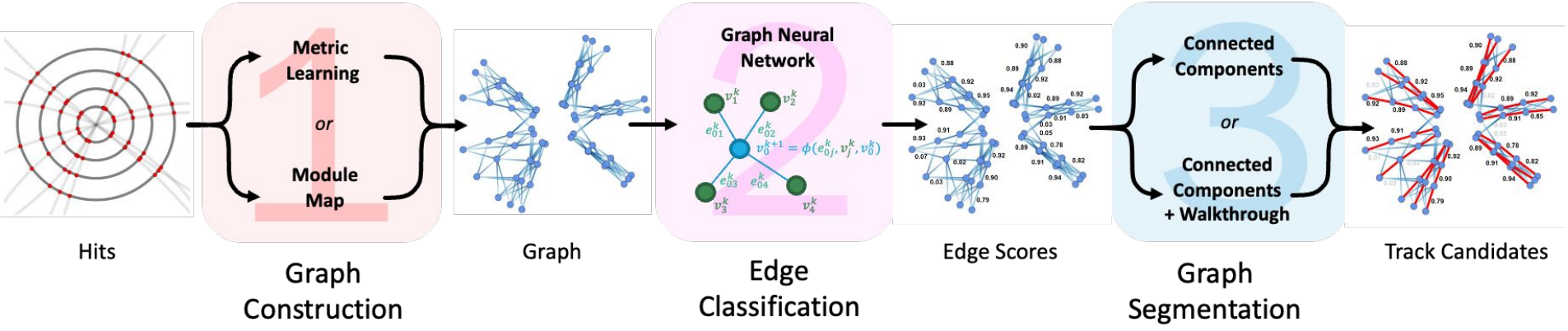


Event reconstruction is a computationally-challenging problem for the (HL)-LHC → Critical element for high-quality physics

- Particle tracking takes ~40% of the reconstruction time

GNNs for Tracking

- Graph Neural Networks (GNNs) are a class of geometric deep learning methods for modeling data dependencies via message passing over graphs
- Detector measurements are represented as nodes. Nodes are associated with each other by learned edges that represent charged particle trajectories



Event Filter GNN Tracking Efforts

Accelerated AI Algorithms for Data-Driven Discovery

Goal: to have a full and optimised GNN pipeline implemented and tested on FPGA

FPGA Strengths over GPUs:

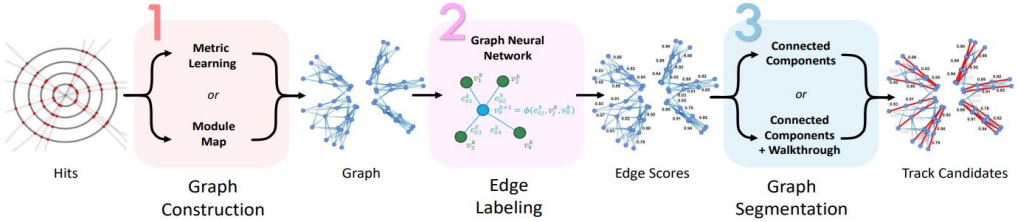
- low-latency inference
- reduced power consumption

Effort to convert python code for firmware implementation:

- Use HLS4ML or FINN (for metric learning and GNN) using ITk data, already explored on TrackML (Elabd et. al (2022))
- Write VHDL code “from scratch” (for module map and final track building)

GNN4ITk pipeline

Since Spring 2022



See: Ju et al. (2021) and CTD (2022) for more info

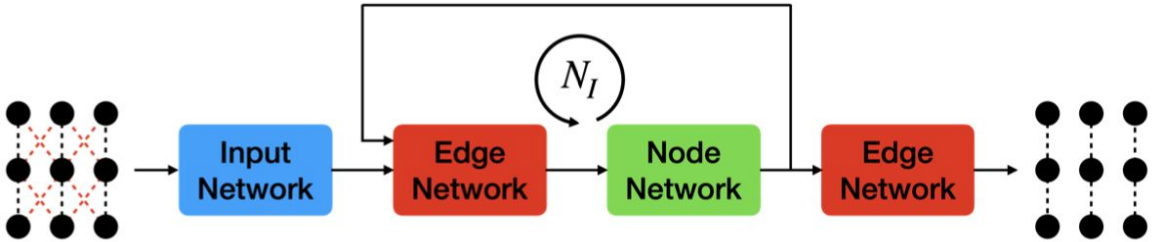
Ongoing and Future Efforts:

- Concerns of limited resources encourages new FPGA studies
- Subgraph construction in regions of detector for module map
- Quantization Aware Training used for metric learning MLP
- Pruning studies on metric learning significant reduction in model size at maintained efficiency (see CHEP (2023))

Hybrid Approach to GNN tracking

- With support from the Illinois Quantum Applications Program, we are working to build on our GNN tracking with classical GNNs and the prior work by others on Quantum GNNs (QGNNs, e.g. C. Tüysüz, et al) to develop a ***Hybrid Neural Network*** (combining classical & quantum networks) approach

QGNN Architecture



Team

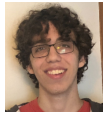
Mark Neubauer
(Prof)

Avik Roy
(Postdoc)

Dewen Zhong
(PhD Student)

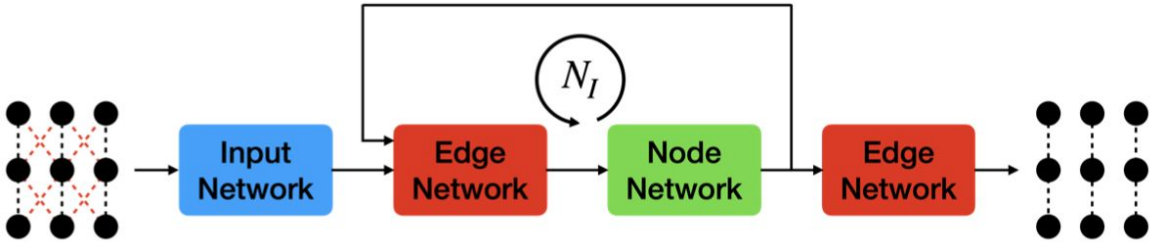
Seung Beom Lee
(Undergraduate)

Mason Camp
Undergraduate

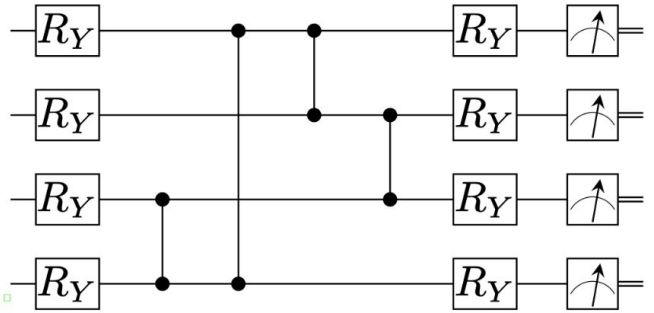


Hybrid Neural Network approach to Tracking

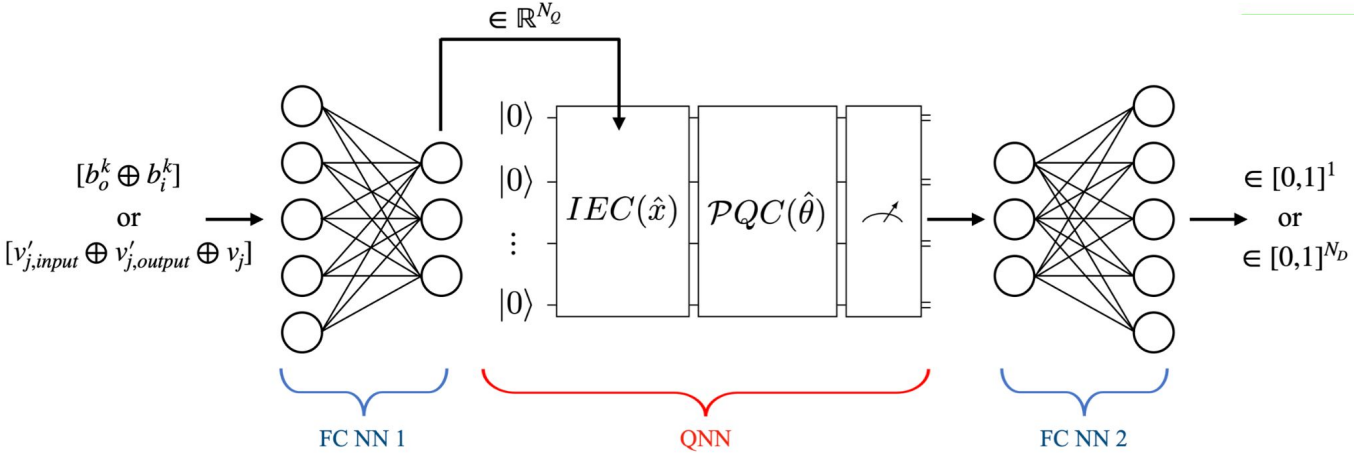
QGNN Architecture



PQC for single layer

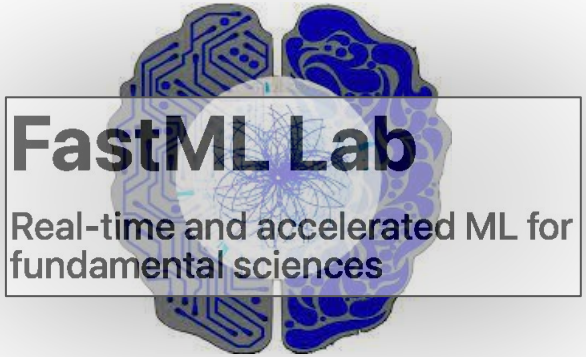


Hybrid NN Architecture



Previous study limited to QC simulations (and up to 16 qubits), mainly due to thousands of circuit executions required by the model

Also no noise included in the simulations



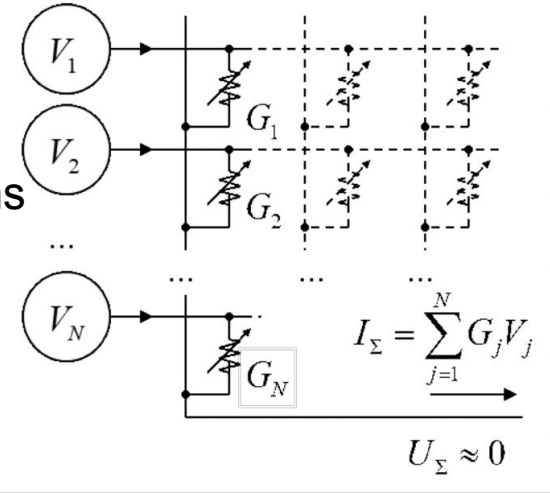
hls4ml is a firmware implementations of machine learning algorithms using high level synthesis language (HLS) in FPGAs with ultra low latency.

- Simplify the hardware implementation process.
- Support for popular machine learning libraries.
- Compatibility with diverse hardware platforms such as FPGAs and ASICs.
- Deploy models in real-time and embedded applications.

Analog AI

Application of analog computing techniques to perform artificial intelligence (AI) tasks.

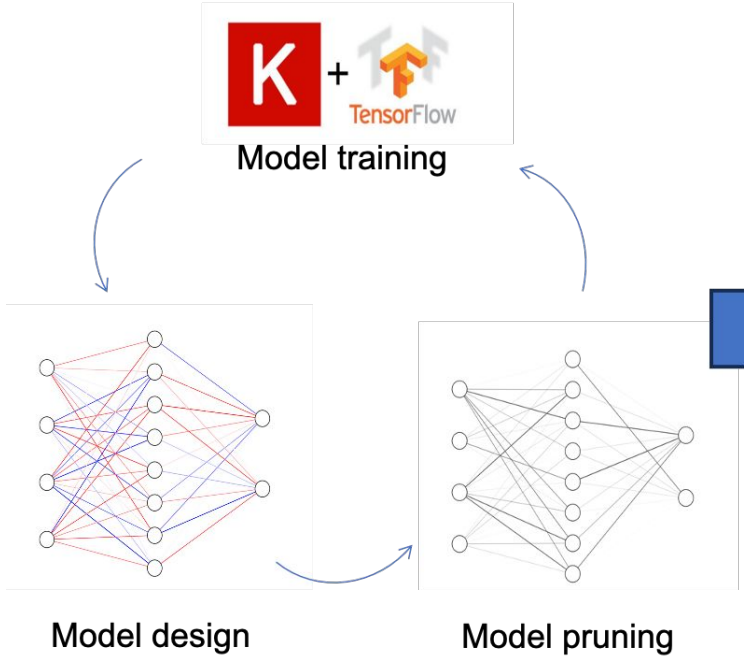
- Analog computing operates on continuous physical quantities, such as voltages or currents.
- Inherent parallelism and efficiency of analog computations
- More energy-efficient, Lower latency to accelerate AI algorithms



Working on new tools for Analog AI in hls4ml

- ***People Involved***: Neubauer, Zhong (UIUC) with collaborators from UIC, FNAL and the Discovery Partners Institute
- In early stage: starting with a simple MLP primitive implemented on a crossbar array
- Synergies with memristors and CMOS devices also being explored

Mixed Analog/Digital Pathway



hls4ml
Internal representations

hls4ml
C code

“Analog” Primitive:
a2d (N,precision)
d2a (N,precision)
Mat-Vec operation

Fab “Primitives”

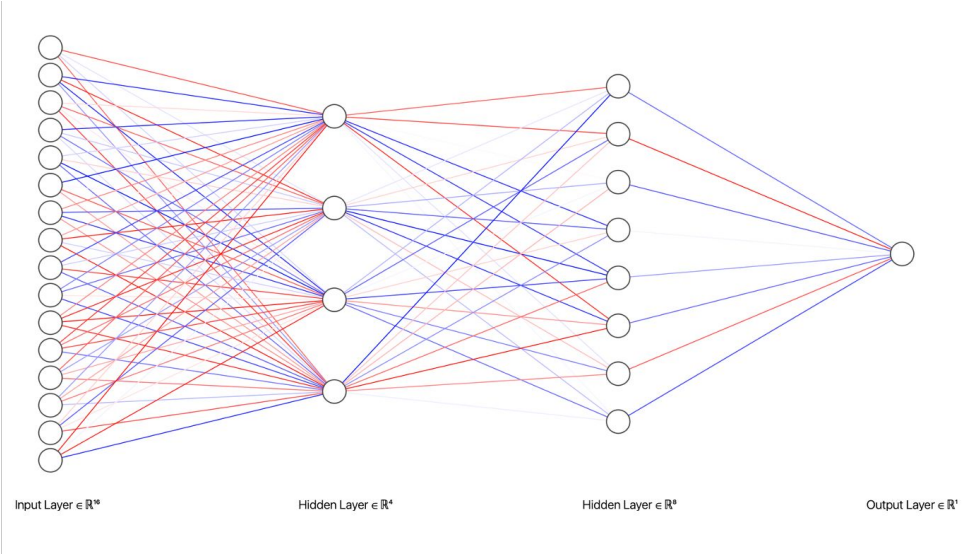
- TSMC Crossbar, 1-bit memristor
- Sonos, Crossbar, 5-bit memristor
- OTA
- FPAA
- 28nm CMOS 1b SRAM “crossbar”

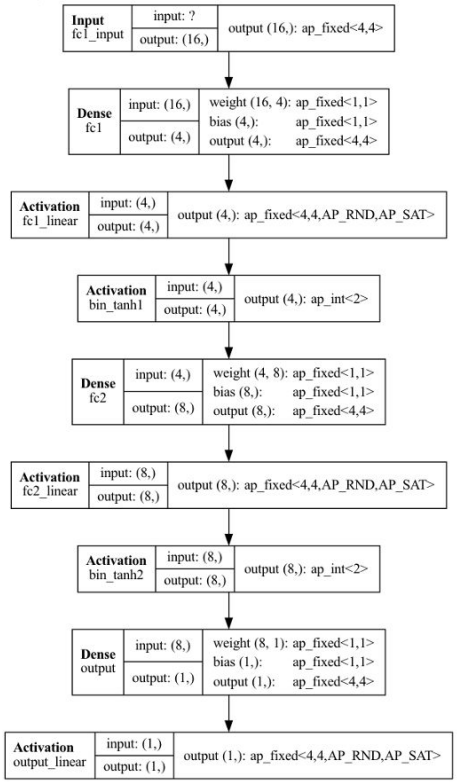
Machine Learning architectures

Accelerated AI Algorithms for Data-Driven Discovery

Custom ANN model

- 16 inputs, with full connecting 4 and 8 nodes Hidden layers and output is 1
- Quantization weight and activation function of NN model and setup output node to 4 bits



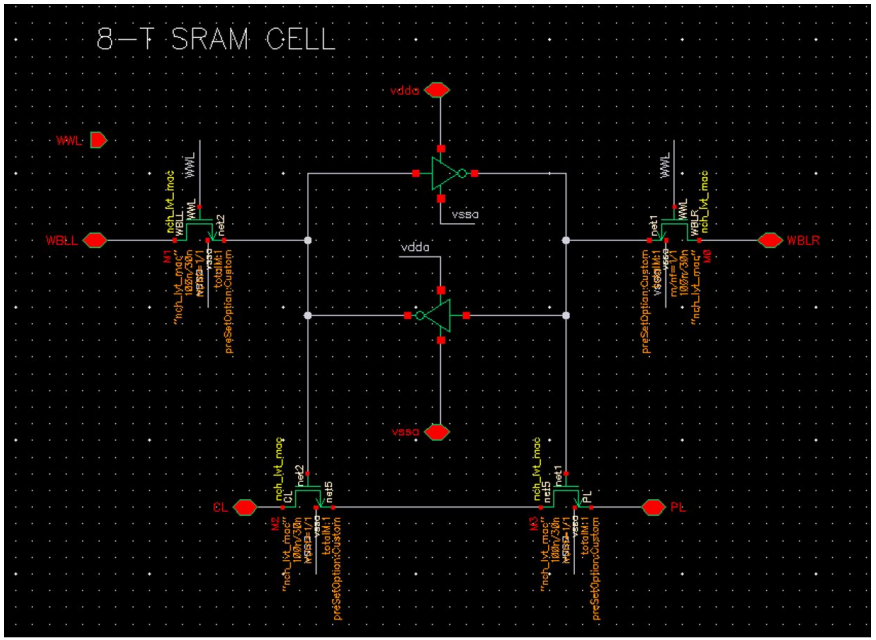


8-T SRAM-Unit Cell

Accelerated AI Algorithms for Data-Driven Discovery

Based off design

- Mf-net: Compute-in-memory sram for multi-bit precision inference using memory-immersed data conversion and multiplication-free operators
- MC-CIM: Compute-in-Memory With Monte-Carlo Dropouts for Bayesian Edge Intelligence



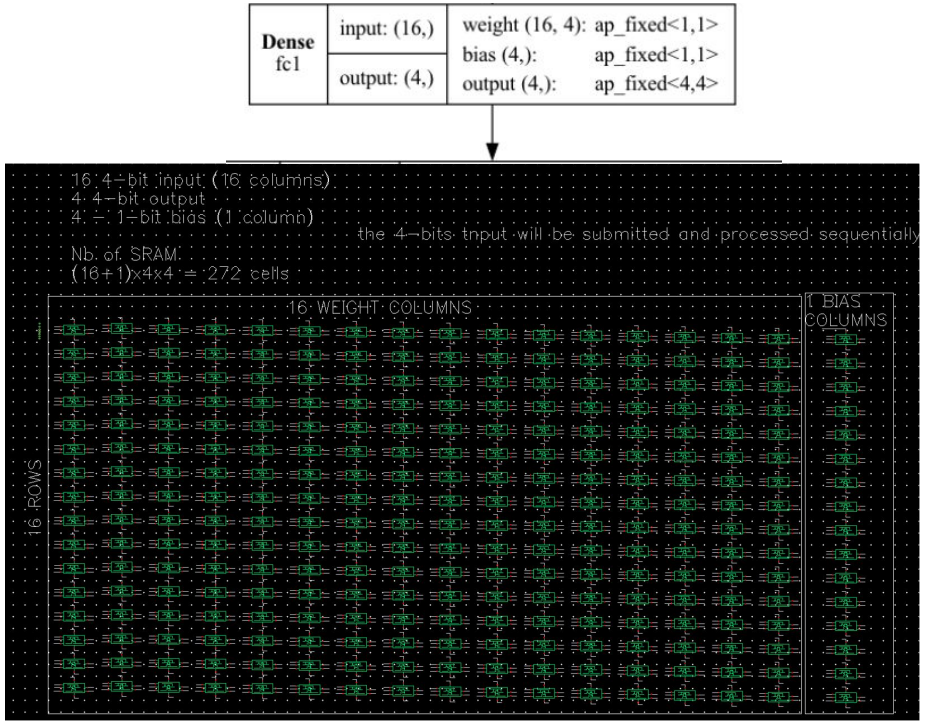
First Layer Analog Representation

Basic Matrix for fc1 layer:

- 16 4-bits input (16 columns)
- 4 4-bits output
- 4 1-bit bias (1 bias column)

All production lines charged are accumulated and summed horizontally bit by bits.

Each bit needs to be binary weight.



Expectation of analog ML and Goal

- Using hls4ml to generate analog AI models and implement analog models to ML-based jet Tagger.
- Provide similar performances as digital hls4ml ML models.

  Further work

- Complete the full analog representation for entire ANN model structure.
- Tuning and Optimize the analog AI model performance for High Energy Physic models and beyond .