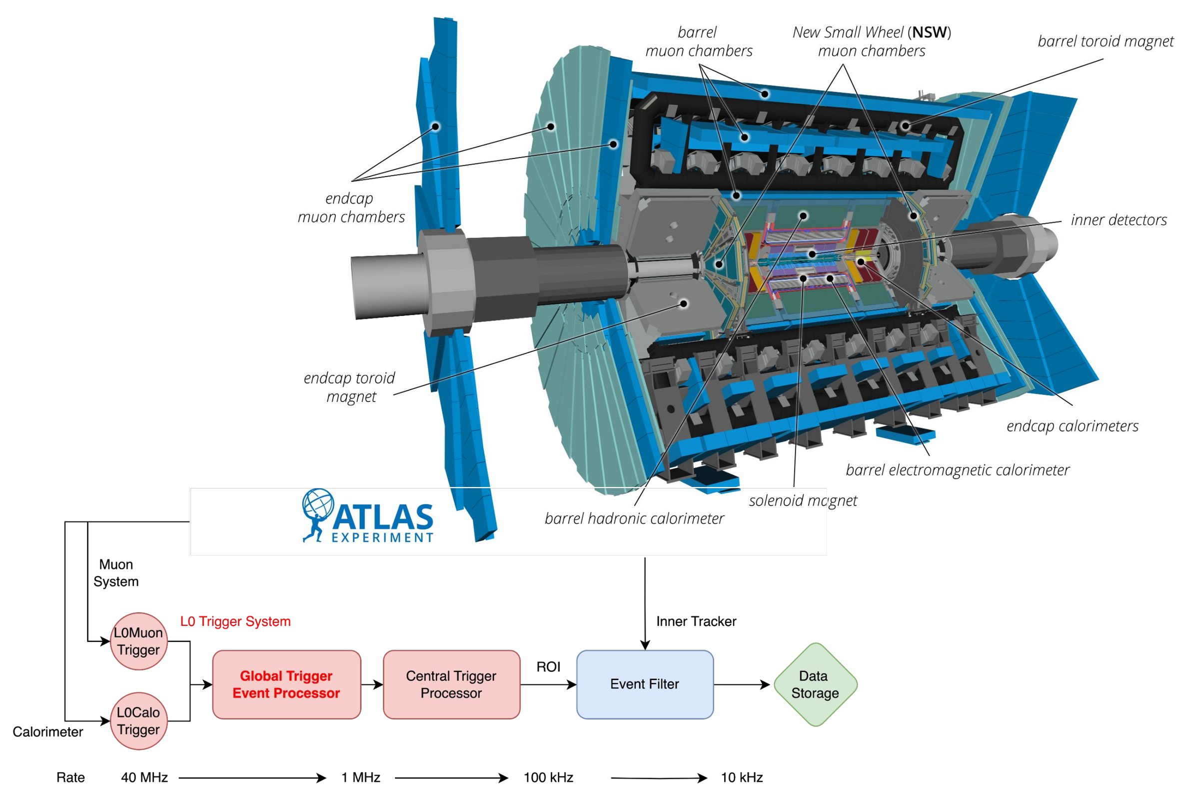
The Bigger, the Better? Optimizing Neural Networks for Calorimeter Calibration in the L0 Trigger with hls4ml



Presenter: Annabel Li[1,2] (ali2@triumf.ca)
Advisors: Kelvin Leong[1,2], Colin Gay[1], Wojtek Fedorko[2], Max Swiatlowski[2]
[1]University of British Columbia, [2]TRIUMF

Background & Motivations

- The Large Hadron Collider (LHC) will be upgraded to High-Luminosity by 2030
  - Pileups will increase: 60 → 200 collisions per bunch crossing
- L0 trigger system (using FPGAs) struggles with current data rate
  - Incorrect calibration in energy deposited -> incorrect events reconstruction[1][2]
  - Low trigger rate discards potentially valuable information

We need a faster, more accurate trigger system.

How Neural Networks Can Help

- DeepSets machine-learning model improves performance in cluster energy regression[1][2]
- 3 stages: **Φ** network, latent space, F network

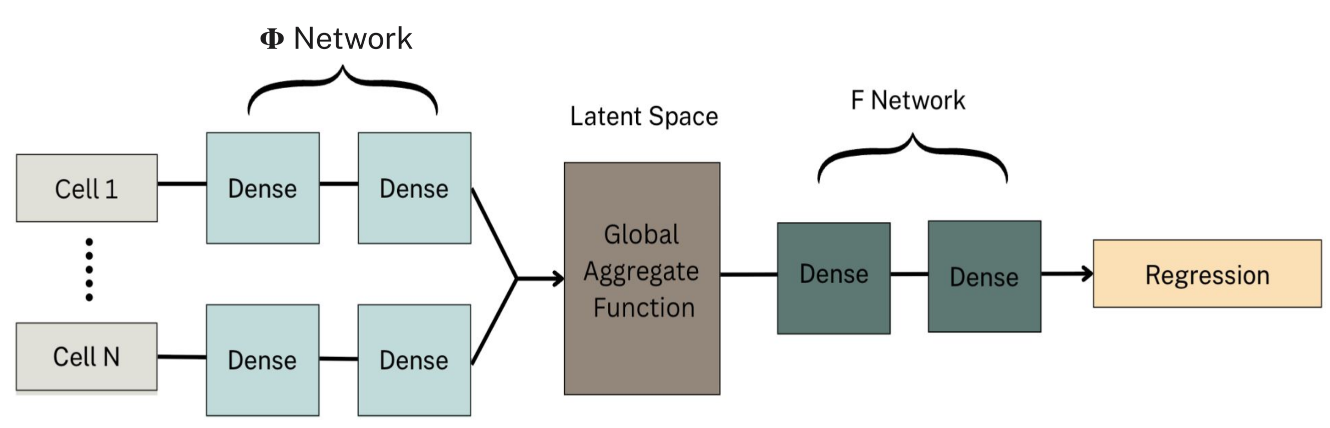


Fig1: Schematic of the DeepSets model

Model architecture:
- 5 phi layers, N nodes each
- Custom sum function (KSum) in the latent space
- 5 F layers, N nodes each

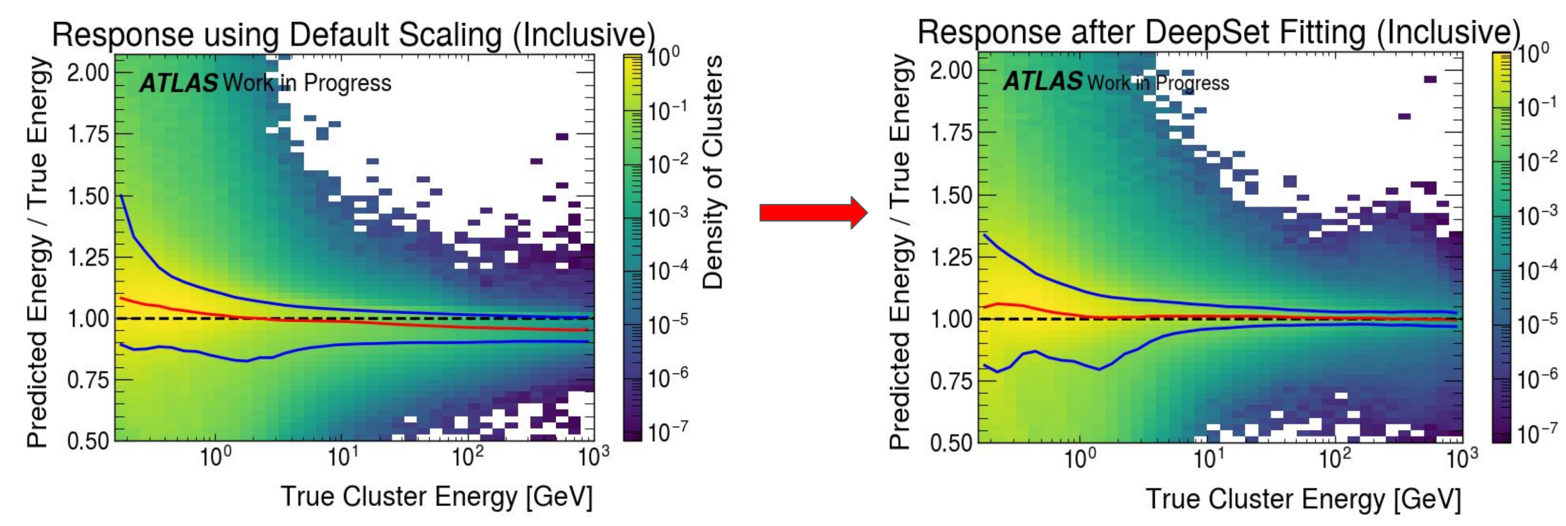
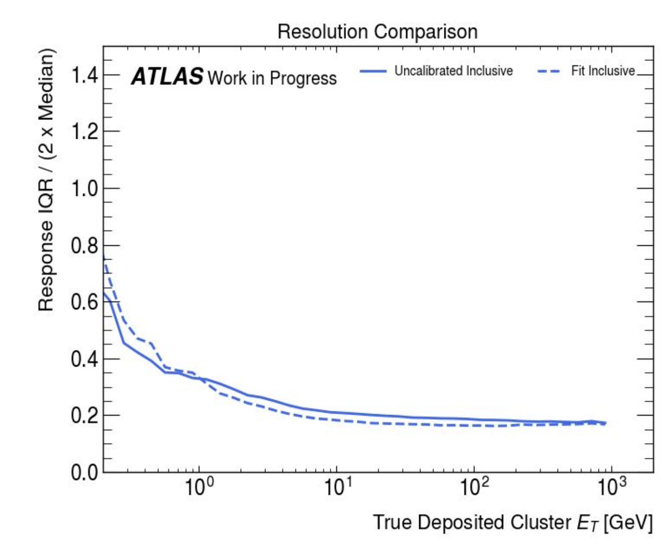


Fig2: trained model results

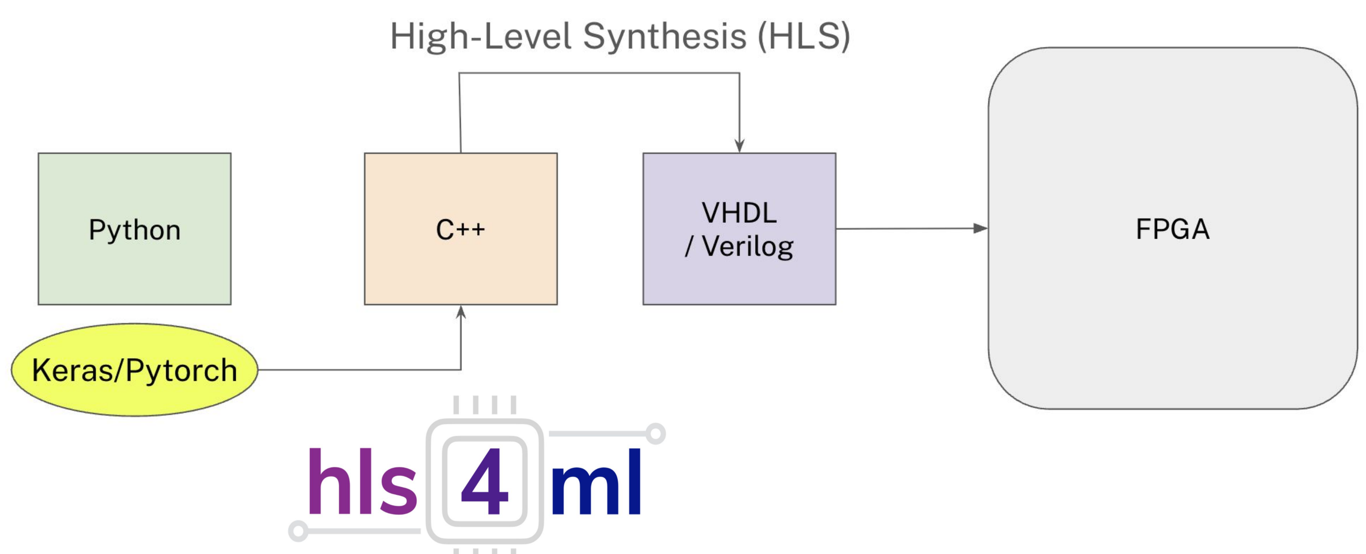
Top: Response from MC samples using default calorimeter calibration (left) vs. DeepSets model (right). Red/blue lines represent the median and IQR responses

Right: Resolution (predictive precision) before vs. after DeepSets fitting



How is Code Implemented on Hardware?

- FPGAs are designed with hardware description languages (VHDL, Verilog)
- The **hls4ml**[3] package automatically converts Python machine learning models into synthesis-ready form



What is Quantization?

- During HLS, floating-point numbers are ***quantized*** to fixed
  - ap_fixed <M, N> = M total bits for the number with N integer bits

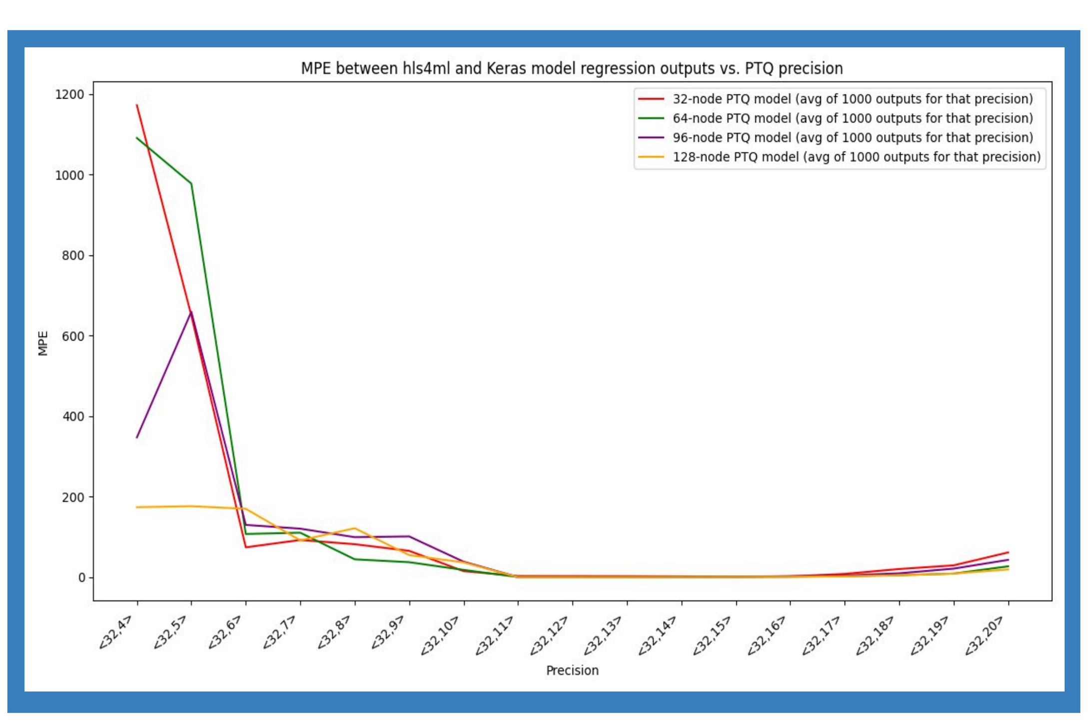
ap_fixed<16,6> → 101101.1010000000 = -18.375

- 2 methods for ML:
  - ***Post-Training Quantization (PTQ)*** -> weights & biases quantized after training
  - ***Quantization-Aware Training (QAT)*** -> model is trained on lower precision operations

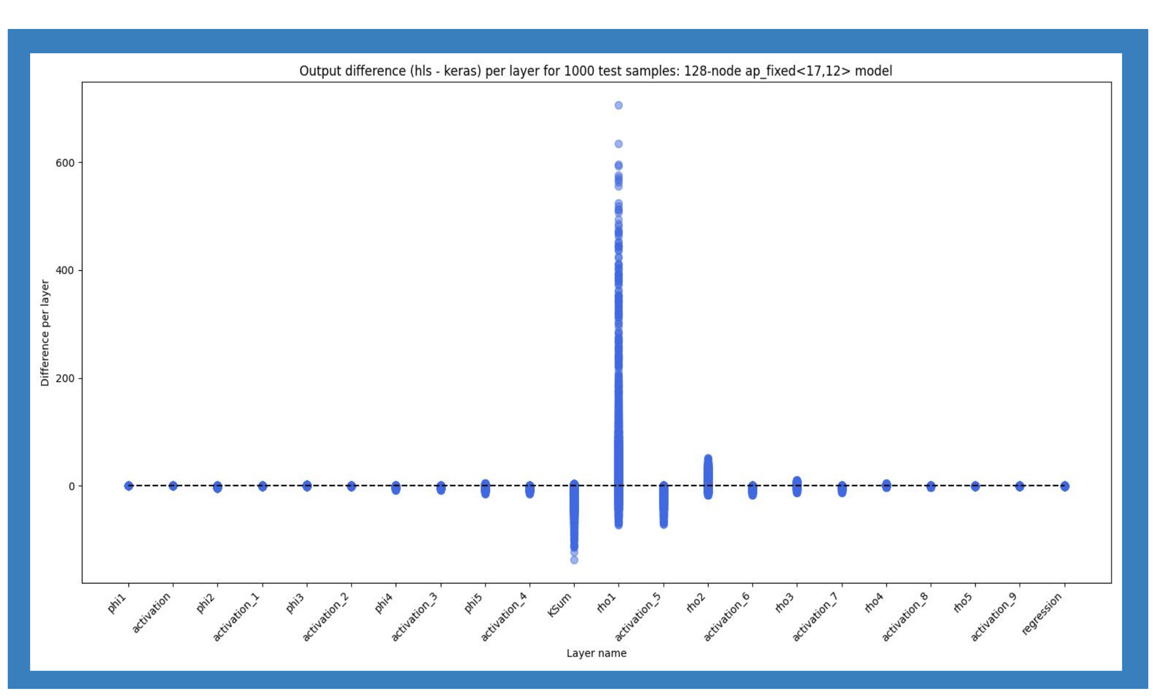
We can use hls4ml to quickly test parameterizations of the DeepSets model for optimization[4][5].

Results - PTQ Analysis

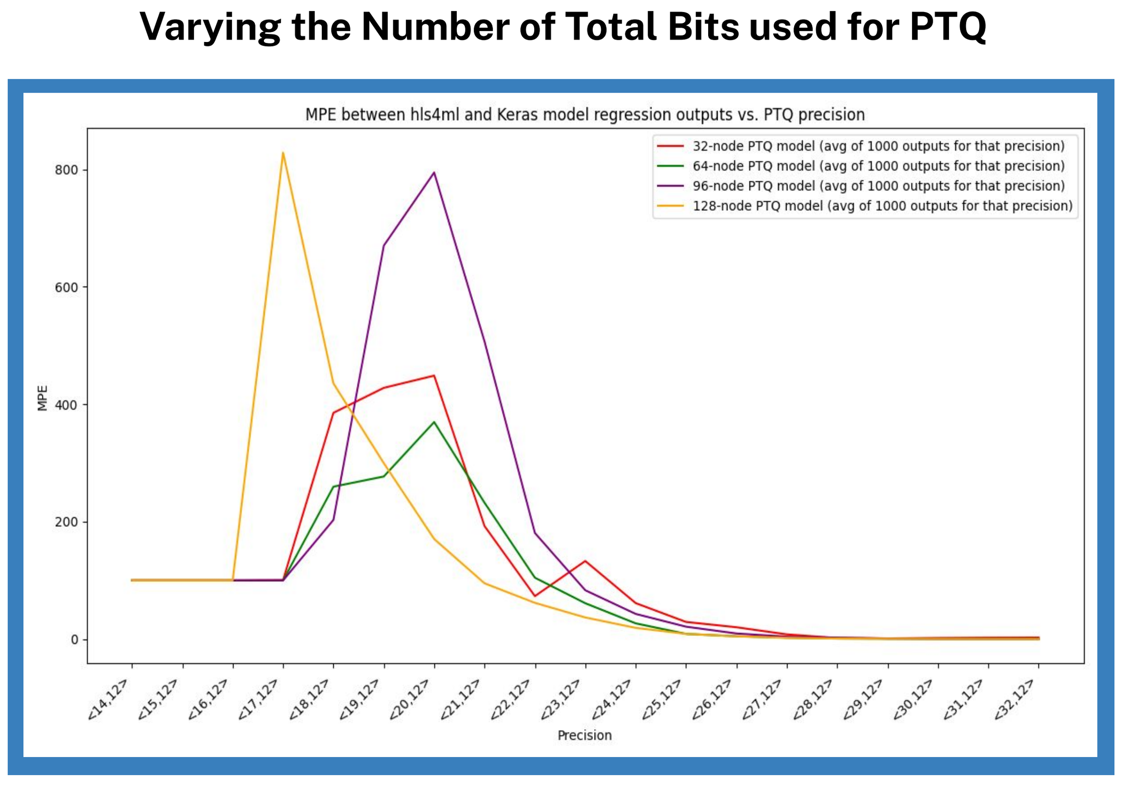
Varying the Number of Integer Bits used for PTQ



Per-Layer Output Difference between hls4ml and Keras



Varying the Number of Total Bits used for PTQ



Optimizing Weights & Biases Precision with PTQ

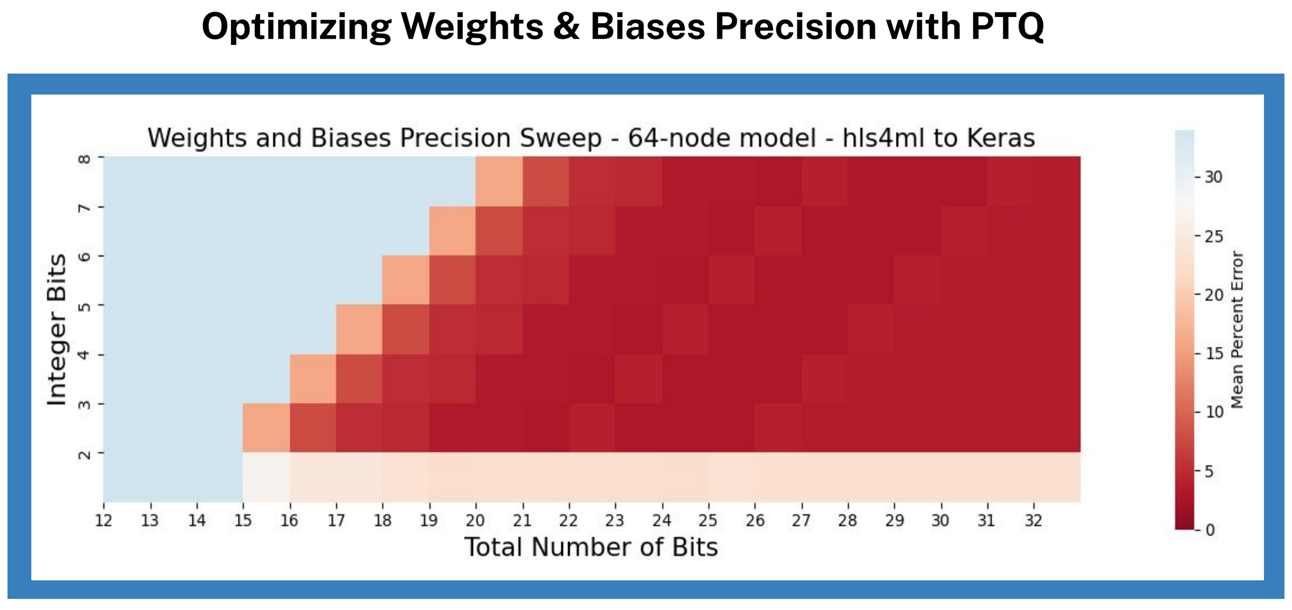


Fig 3, clockwise from top left:
- MPE from Keras regression output, varying integer bits
- MPE from Keras regression output, varying total bits
- Per-layer output differences between Keras and HLS model
- Weights and biases precision sweep, keeping intermediate and final vector outputs as ap_fixed<32,14>.

Conclusion & Next Steps:

Problem: Current hardware system at the LHC is unsuitable for the HL upgrade

Project goal: Optimize ML model size and precision for L0 trigger deployment

Findings:
- Larger models deviate more significantly from their equivalent Keras model at lower precisions
- More integer bits is needed to maintain KSum layer accuracy

Next Steps:
- ❏ Explore further optimization strategies
  - ❏ QAT
  - ❏ Adjusting intermediate and final vector output precisions

Reference List:
[1] "Deep Learning for Pion Identification and Energy Calibration with the ATLAS Detector," tech. rep., CERN, Geneva, 2020.
[2] "Point Cloud Deep Learning Methods for Pion Reconstruction in the ATLAS Experiment," tech. rep., CERN, Geneva, 2022.
[3] F. Fahim, et al., "hls4ml: An open-source codesign workflow to empower scientific low-power machine learning devices," March 2021.
[4] P. Odagiu, et al., "Ultrafast jet classification at the hl-lhc," Machine Learning: Science and Technology, vol. 5, p. 035017, July 2024.
[5] C. Antel, "QDIPS: Deep Sets Network for FPGA investigated for high speed inference on ATLAS," tech. rep., CERN, Geneva, 2025.