PicoCal and ML methods for calorimetry

29 January 2026

Christina Agapopoulou[1], Gaelle Khreich[1], Aniol Lobo Salvia[2], Jean-François Marchand[2], Stéphane T'Jampens[2], Guillaume Vouters[2]

[1] IJCLab, Orsay (Fr)
[2] LAPP, Annecy (Fr)

ODISSEE
Online Data Intensive Solutions for Science in the Exabytes Era

Laboratoire de Physique des 2 infinis Irène Joliot-Curie

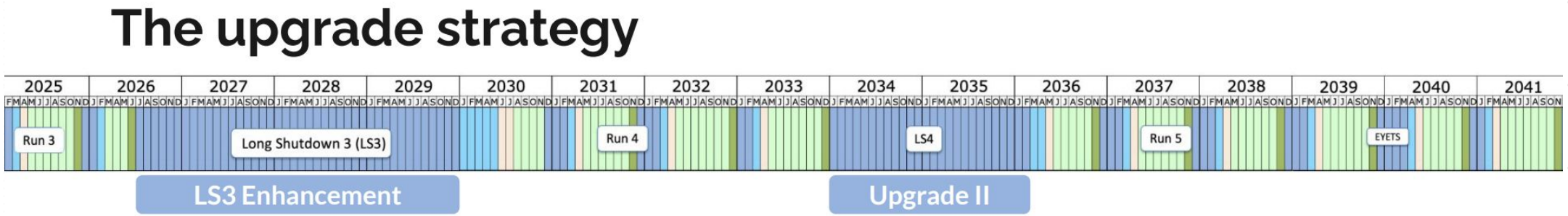
LAPP Laboratoire d'Annecy-le-Vieux de Physique des Particules

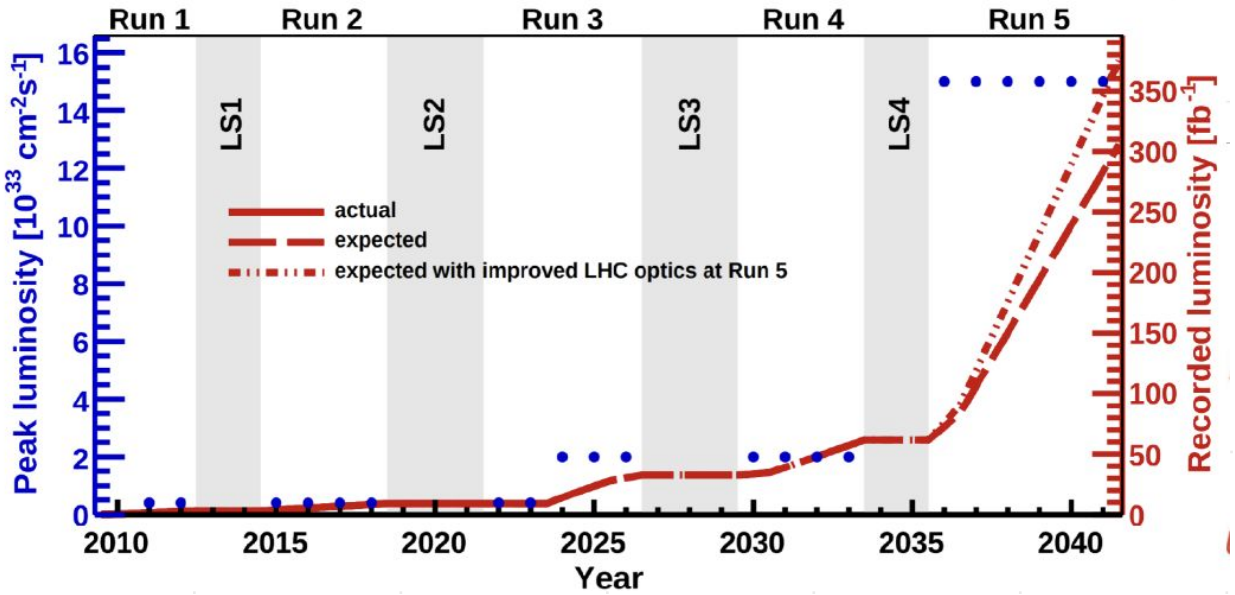
Outline

- Introduction:
  - The challenge
  - LHCb Upgrade II
  - PicoCal
- Goals and strategy
- Current calo reconstruction algorithms
- Future calo reconstruction algorithms
- Status
- Conclusions

The Challenge

- LHCb has an ambitious program to increase the recorded integrated luminosity

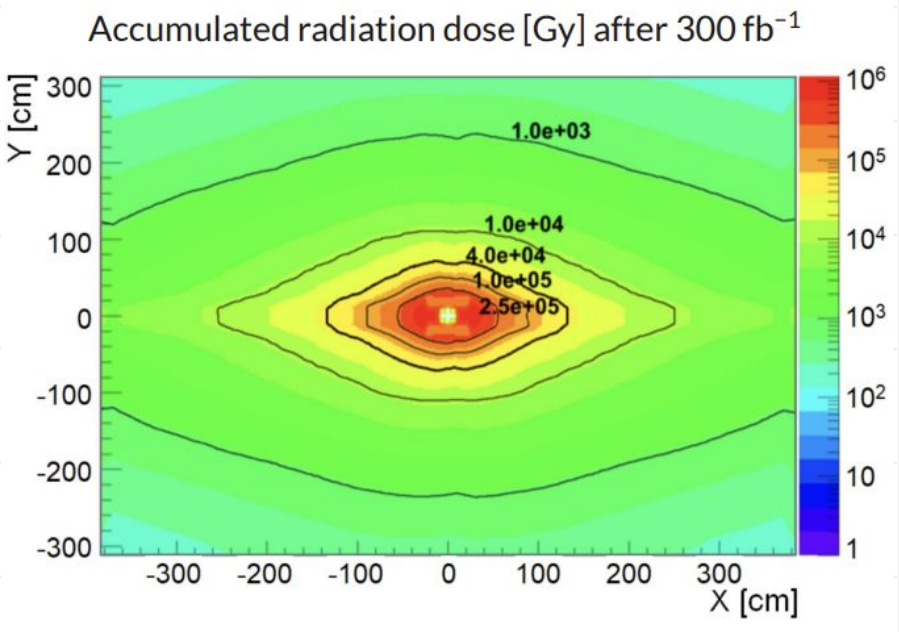




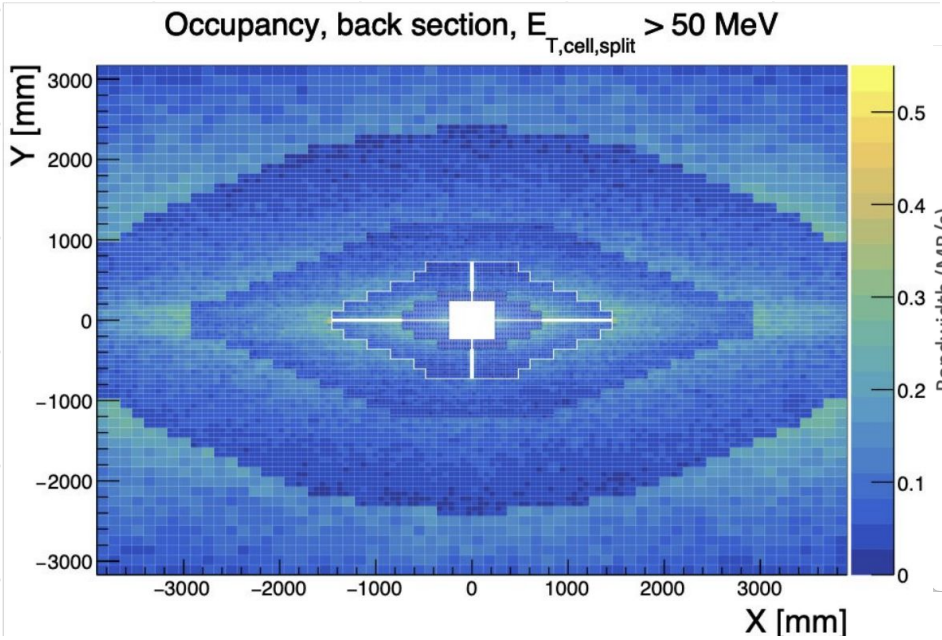
The Challenges

- This comes with a series of challenges

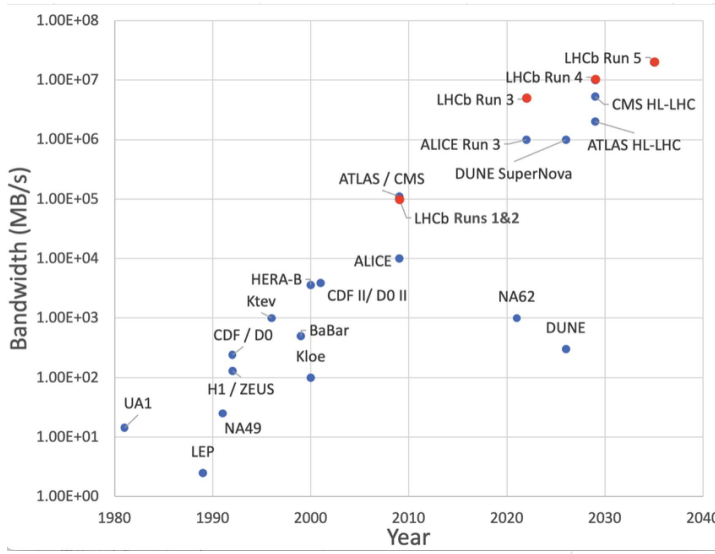
Radiation dose



Pile-up, Occupancy

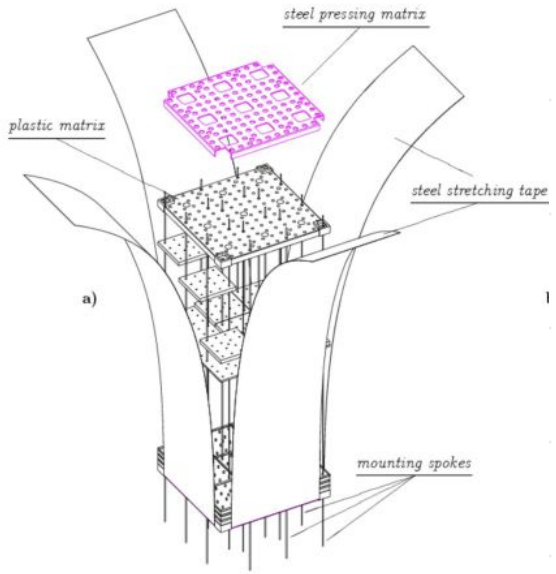


Bandwidth, throughput

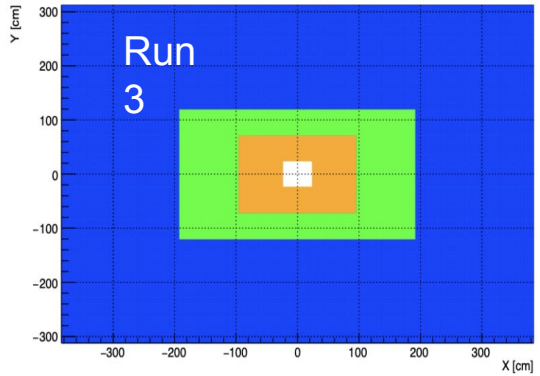


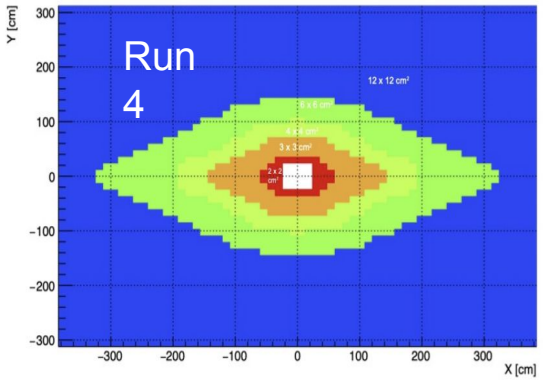
The Solutions

SpaCal technology

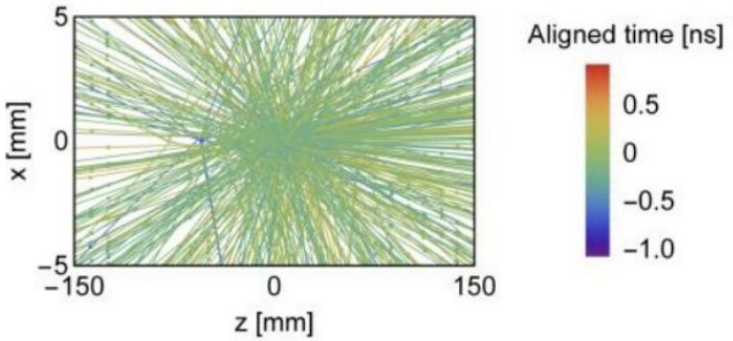


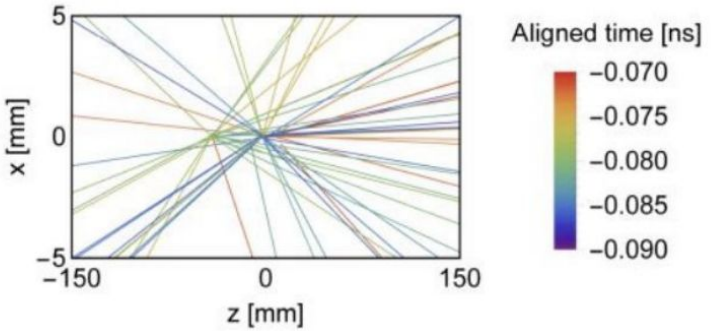
Better granularity





O(10ps) Timing





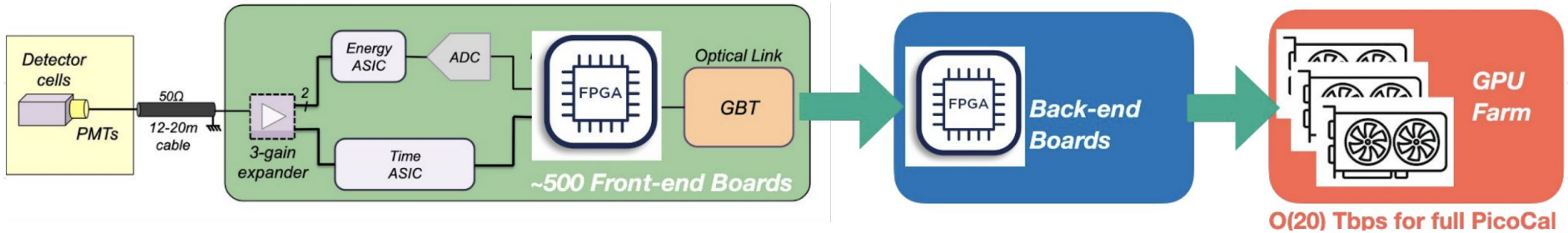
Higher pile-up, higher granularity, timing information… how will this translate to the throughput requirements?

The general idea of this work

- ML is becoming an increasingly used tool in HEP, but in past mostly for online classification, or offline - using it in high-throughput reconstruction applications is the next step
- Main candidate for LHCb's HLT reconstruction in U2 are GPUs
- But we should also not close the doors to other processors / accelerators - the market is quite volatile at the moment, we need to benchmark different possibilities with our use case
- FPGAs are particularly attractive, because we already have them for DAQ
  - potentially spare compute resources
  - already exploited in Run 3 (velo RETINA clustering)
- We want to study ML applications for reconstruction in a high-throughput heterogeneous environment, benchmark different accelerators, and, in the long term understand how we can build a DAQ - HLT system that gives the best performance, computing power and energy efficiency
- PicoCal relatively "stand-alone" detector: a good playground for this kind of study

Goals

Optimization of the PicoCal processing chain

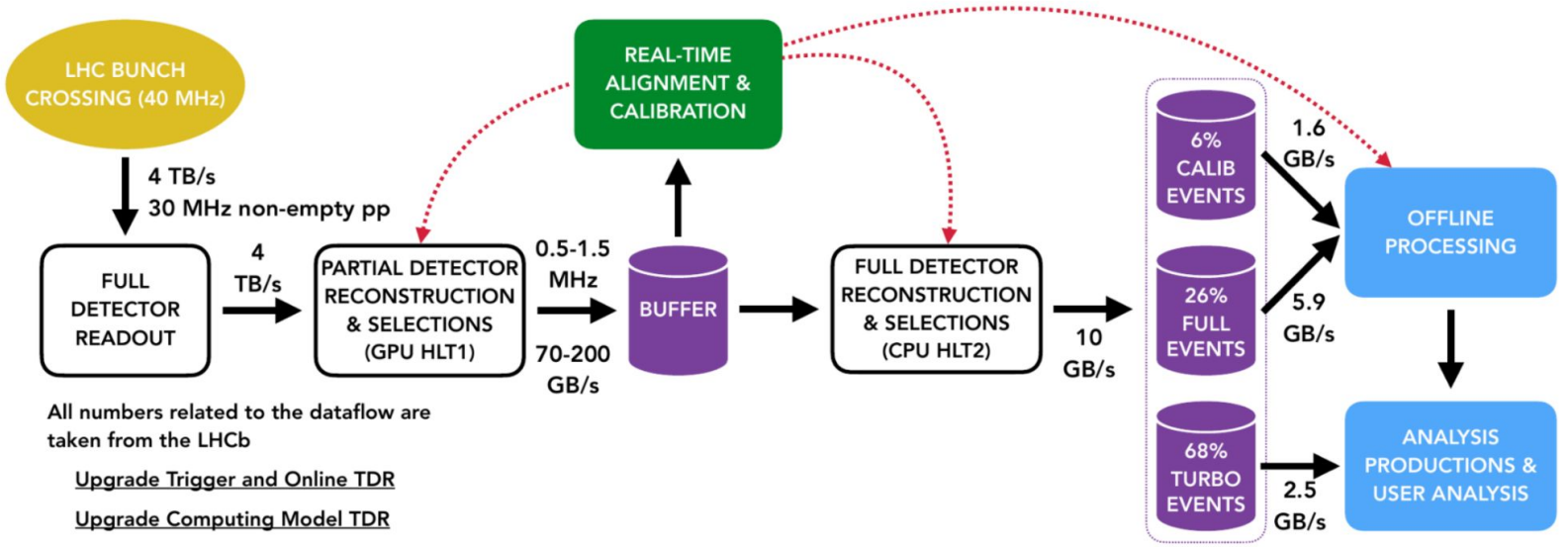


Roadmap

- Start by implementing the current HLT1 reconstruction algorithm to PicoCal simulated data
- Explore AI models in high-throughput conditions
- Add timing
- FPGA acceleration (e.g. seeding) (more far in the future, not covered here)

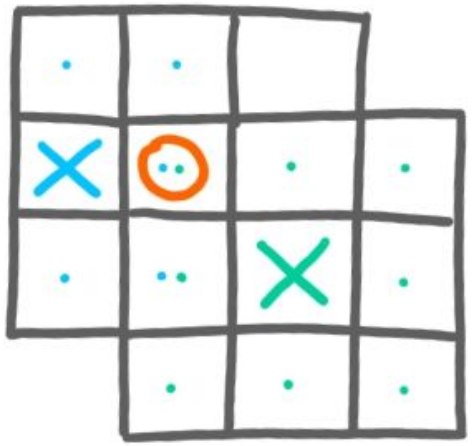
Run 3 trigger

- Two steps “real-time” reconstruction and selection
- Reminder: HLT2 performs offline-quality reconstruction, no reconstruction happening offline!



ECAL reconstruction in current HLT1

- Finding local maxima
- Clustering on 3x3 cells with overlap corrections



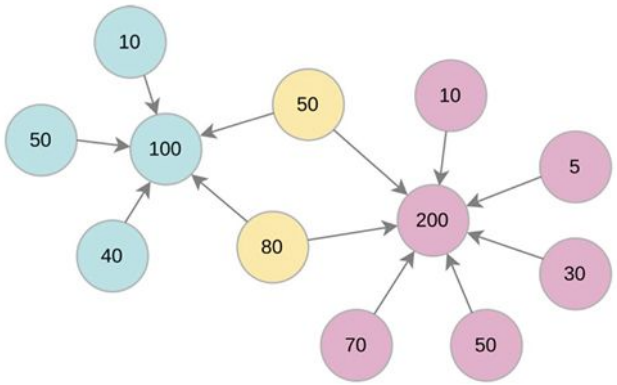
$$correction_{cluster1} = E_{overlap} \frac{E_{cluster2}}{E_{cluster1} + E_{cluster2}}$$

ECAL reconstruction in current HLT2

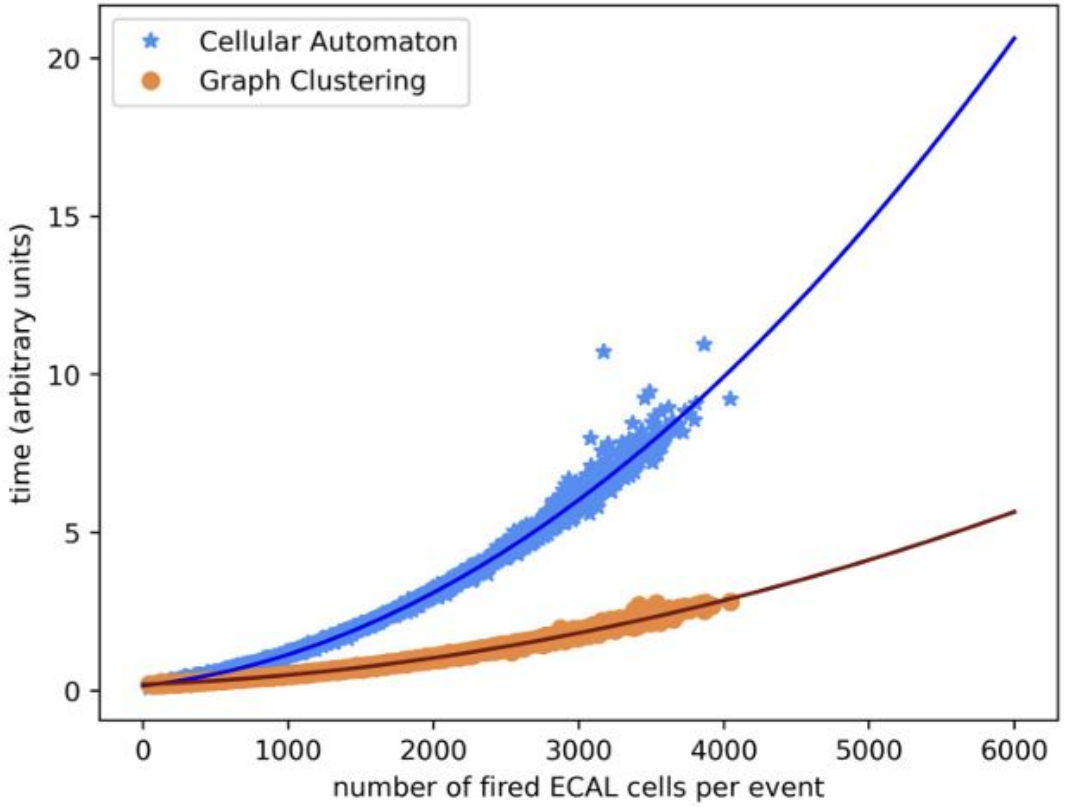
- Graph Clustering

			10	5
	10	50	200	30
50	100	80	70	50
	40			

two clusters with overlapping cells on the calorimeter



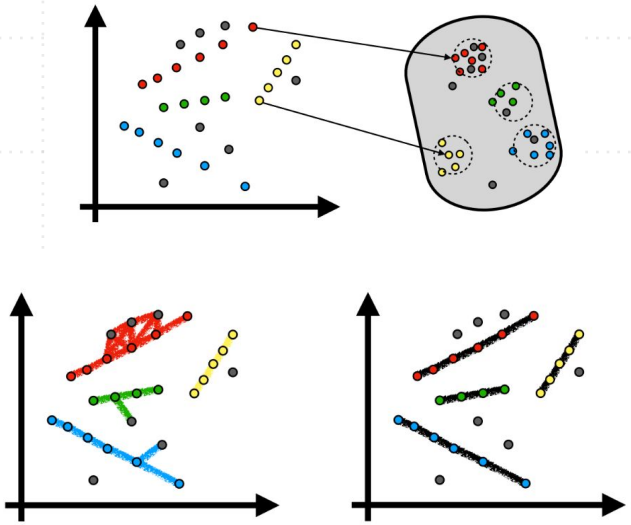
their graph representation



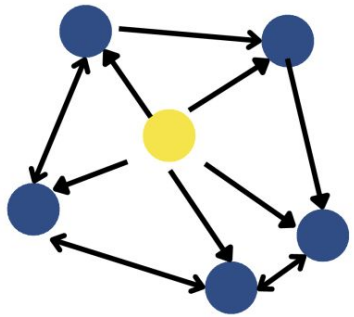
Upgrade II developments

- Big interest in the community in ML implementation
  - In particular GNNs

Tracking GNN:
ETX4VELO



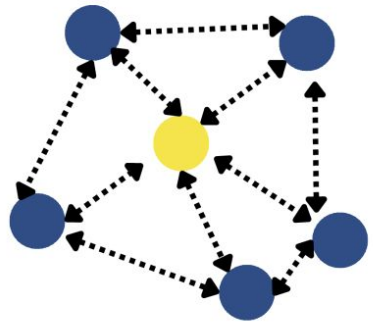
PicoCal GNN:
FMPGNN



Full Message Passing

Inspiration from CMS:
- distance-weighted graph networks
- FPGA implementation

PicoCal GNN:
GARNET



GarNet

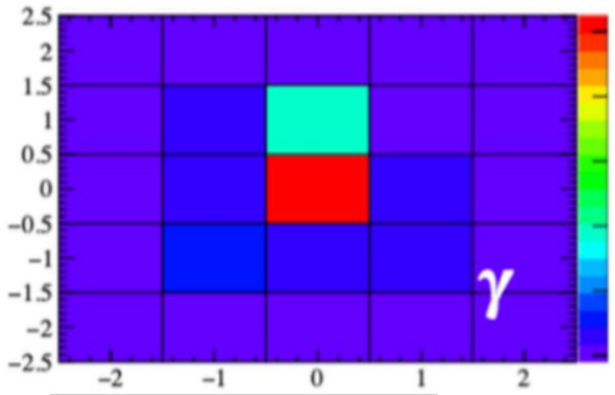
LPNHE PARIS

Upgrade II developments: GNNs for PicoCal reconstruction

Data preprocessing

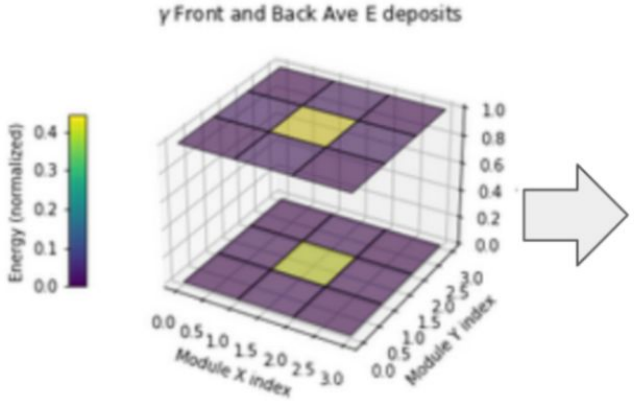
→ Spacal Simulation with Single Photons (particle gun) and minbias clusters
→ Raw PicoCal Data converted to KNN-based graph → node (E, position), edge (spatial links), and global (seed position) features

$\mathcal{L}_{\text{beam}} = 1.5 \times 10^{-34} cm^{-2} s^{-1}$

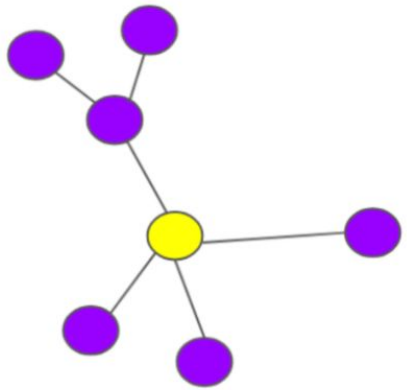


$0.5 \leq E_T \leq 5\ \text{GeV}$

SpaCal Simulation



Choose 3 x 3 clusters w/ Front and Back Energies



Graph Inputs:
Nodes: Etot/cell, Efront, Eback, x, y, relative position to seed
Edges: Δx, Δy, ΔE, dij
Global: Etot/particle

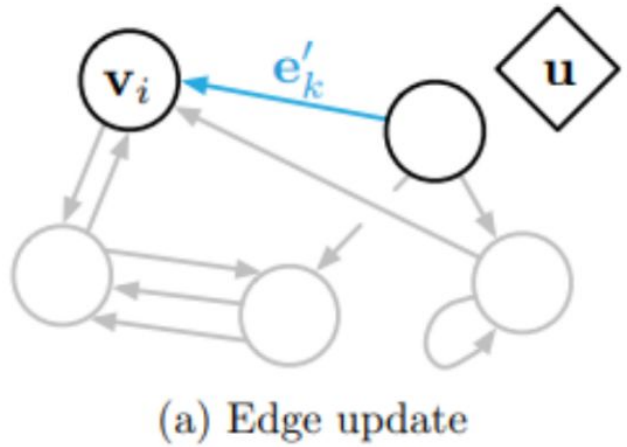
100k in FULL ECAL, 12k in the Spacal Region

Upgrade II developments: GNNs for PicoCal reconstruction

Slides from F. Souza

- Nodes, edges, and globals are updated through aggregation and MLPs

BLUE updated by BLACK (not utilizing GREY)



(a) Edge update

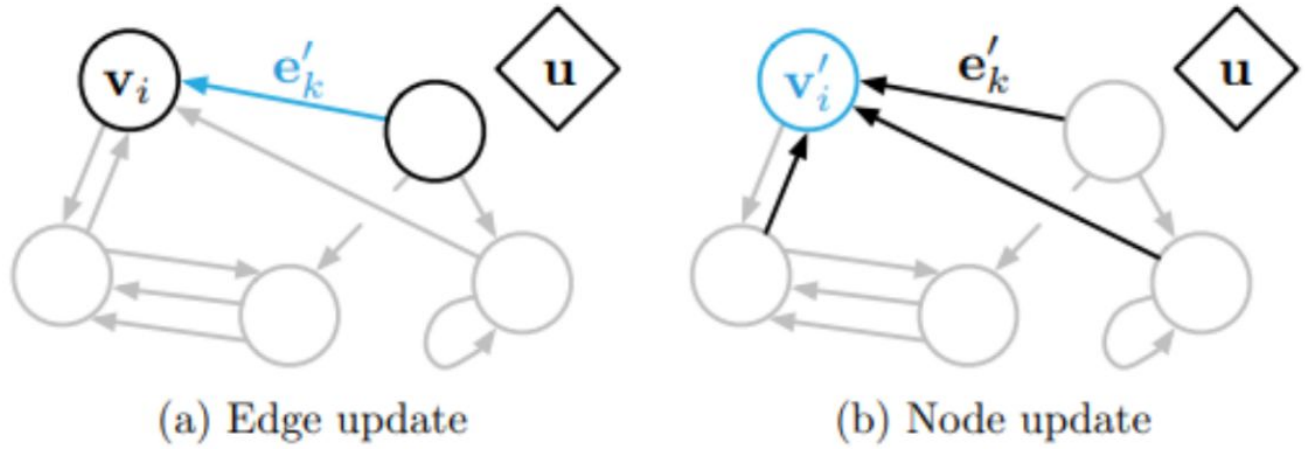
Update edges using connected nodes and globals

Upgrade II developments: GNNs for PicoCal reconstruction

Slides from F. Souza

- Nodes, edges, and globals are updated through aggregation and MLPs

BLUE updated by BLACK (not utilizing GREY)



(a) Edge update

Update edges using connected nodes and globals

(b) Node update

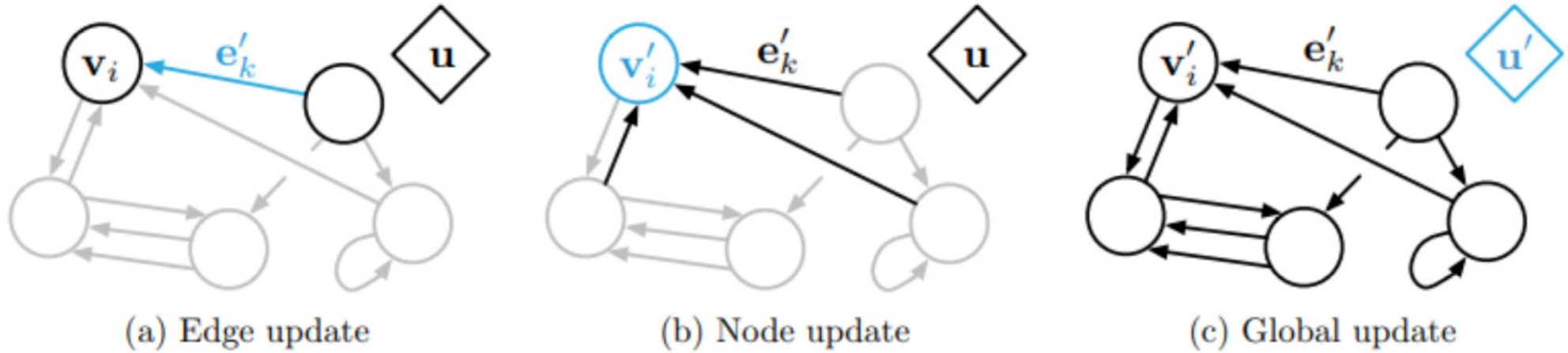
Update nodes using (new) edges and globals

Upgrade II developments: GNNs for PicoCal reconstruction

Slides from F. Souza

- Nodes, edges, and globals are updated through aggregation and MLPs

BLUE updated by BLACK (not utilizing GREY)



(a) Edge update

Update edges using connected nodes and globals

(b) Node update

Update nodes using (new) edges and globals

(c) Global update

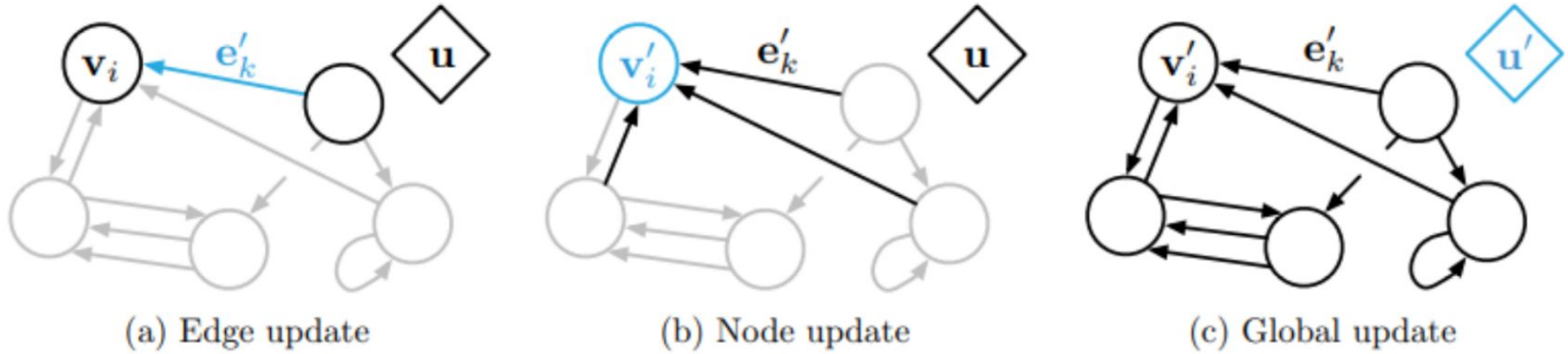
Update globals using (new) nodes and (new) edges

Upgrade II developments: GNNs for PicoCal reconstruction

Slides from F. Souza

- Nodes, edges, and globals are updated through aggregation and MLPs

BLUE updated by BLACK (not utilizing GREY)



(a) Edge update

Update edges using connected nodes and globals

(b) Node update

Update nodes using (new) edges and globals

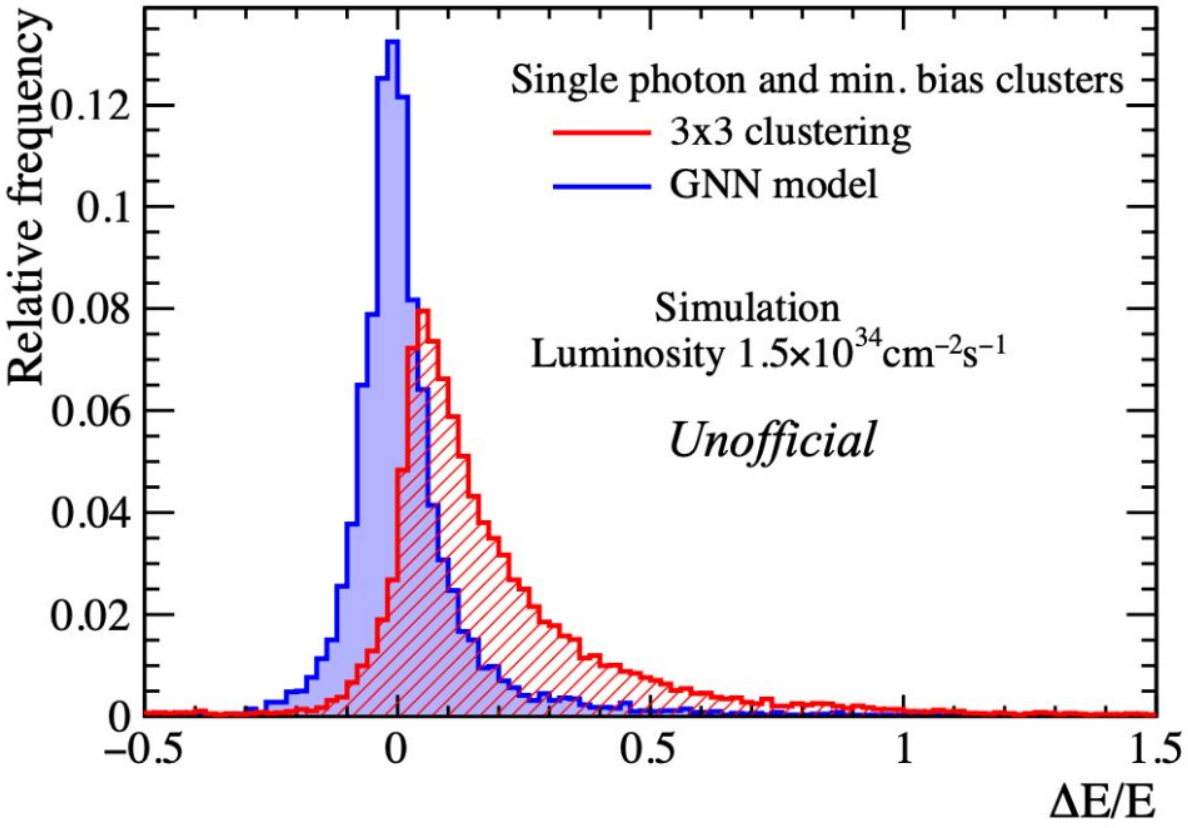
(c) Global update

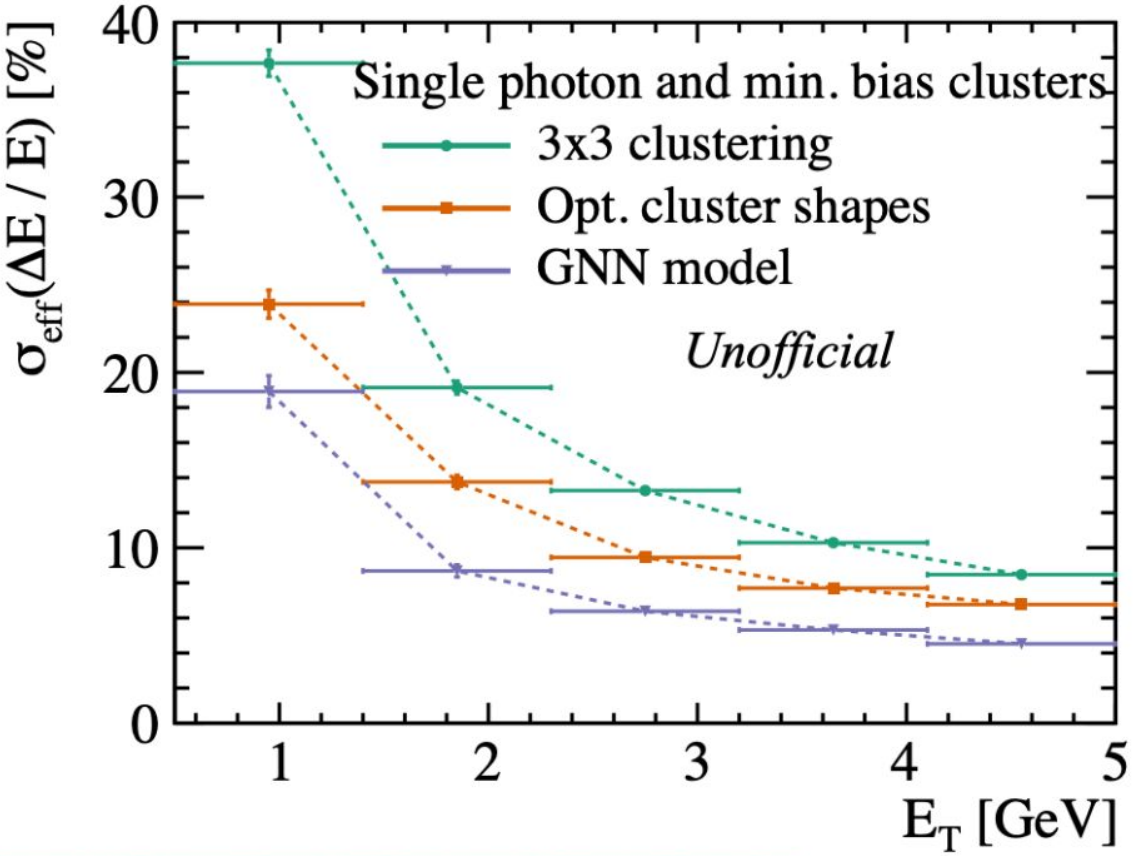
Update globals using (new) nodes and (new) edges

This process can be repeated many times
Each iteration can have a unique set of NNs

Upgrade II developments: GNNs for PicoCal reconstruction

Slides from F. Souza





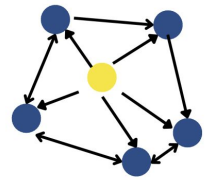
The GNN outperforms the standard approaches over the full E_T range

Upgrade II developments: GNNs for PicoCal reconstruction

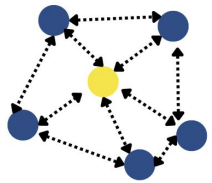
Slides from U. Perez

GarNet

- A lightweight, attention-enhanced variant GNN architecture
  - Introduced for real-time particle reconstruction at CMS by Iiyama et al. (Eur. Phys. J. C 79, 608 (2019); Front. Big Data 3, 598927 (2021))
  - Explicit edges are replaced by learnable aggregators connecting nodes through latent vertices
    - This reduces the number of mathematical operations, decreasing training and inference time



Full Message Passing

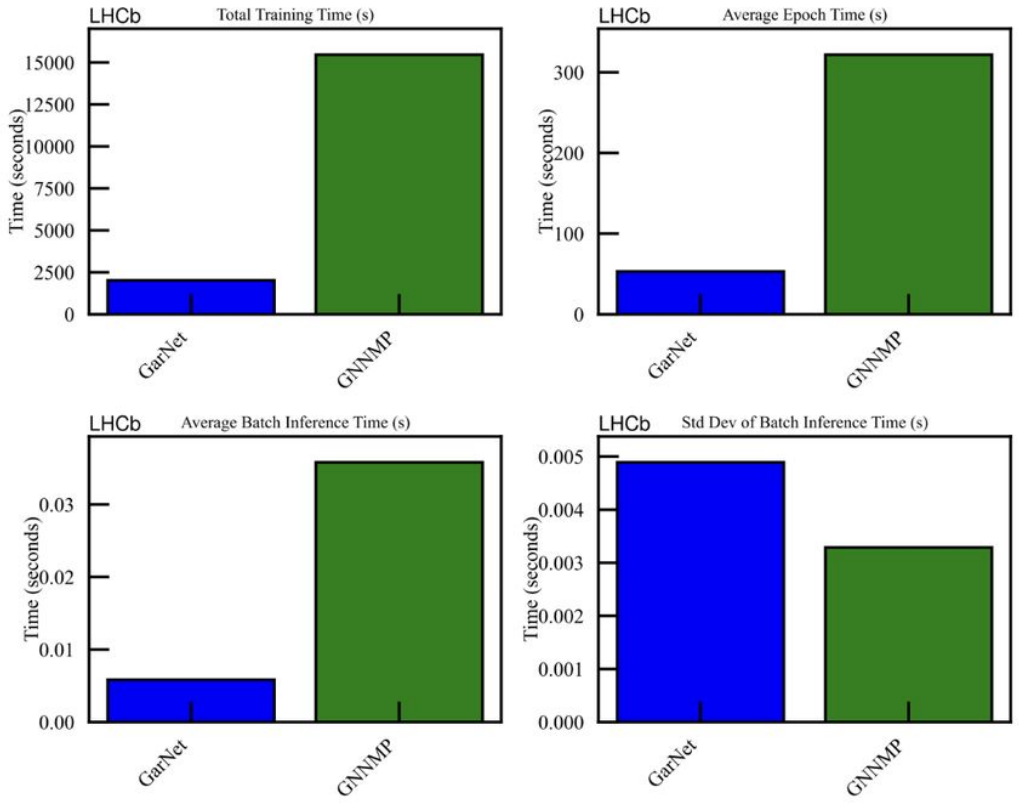


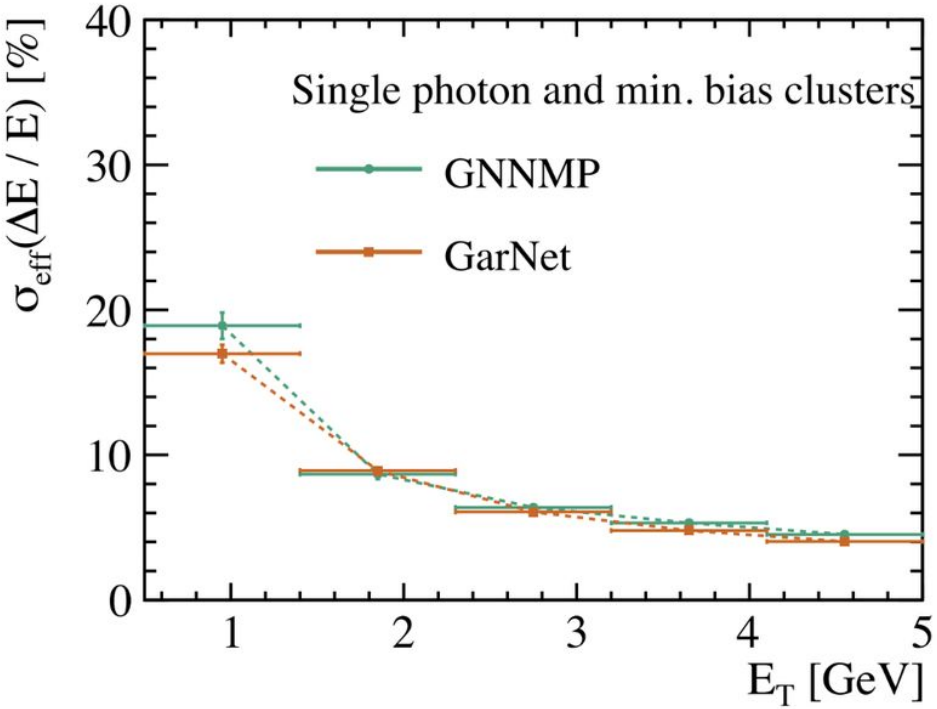
GarNet

☑ GarNet has a know ONNX implementation for FPGA deployment

Upgrade II developments: GNNs for PicoCal reconstruction

- GarNet has similar performance, but is 6-7x faster!





Upgrade II developments: GNNs for PicoCal reconstruction

Extra developments are ongoing

- Distillation and quantization
- ONNX Runtime implementation

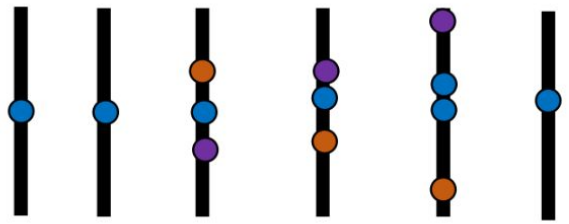
Some lessons from tracking: ETX4VELO

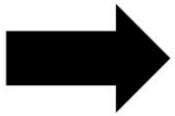
Motivations

Graph Neural Network (GNN)-based track-finding pipeline based on the work of **Exa.Trkx** (*Eur. Phys. J. C* **81**, 876 (2021))

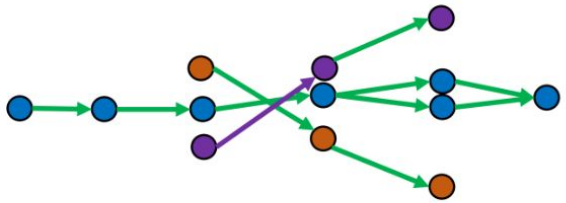
From A. Correia's presentation, CTD23

- Demonstrated **near-linear** inference time w.r.t. # hits
  - *Conventional* algorithms are worse-than-quadratic
  - Increase in instaneneous luminosity in future upgrades over the next decade
    → need for **even more high-throughput** track-finding algorithms
- **High-parallelisation** potential → compatible with current **GPU-based Allen** trigger
- Future implementation in Allen ⇒ allow **like-for-like comparison** with conventional algorithms
- Representation of tracks with a graph quite *natural*



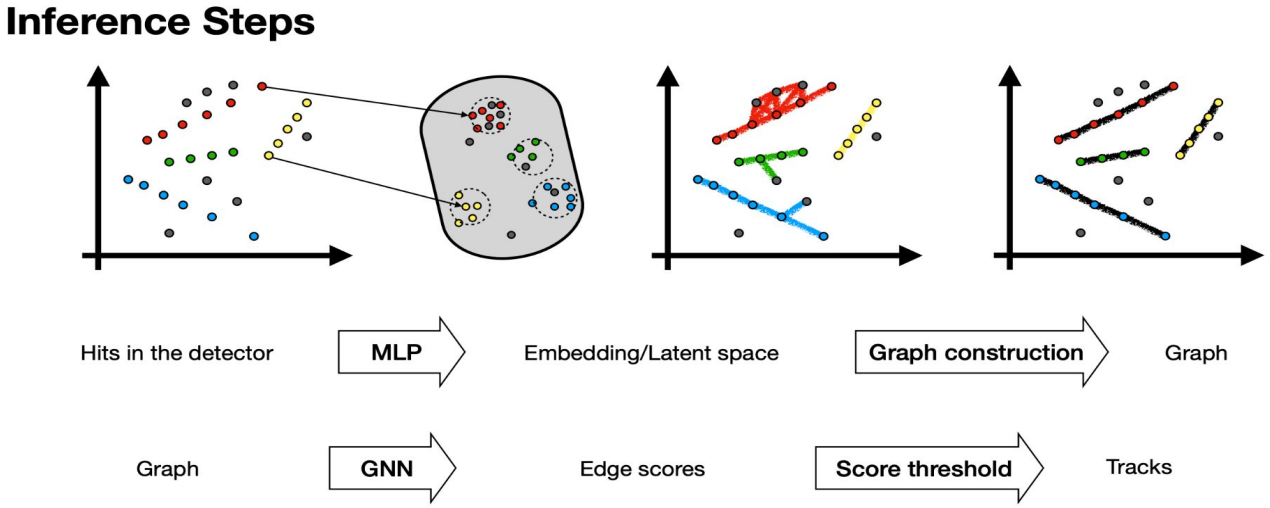


Pure graph representation



GPU Implementation

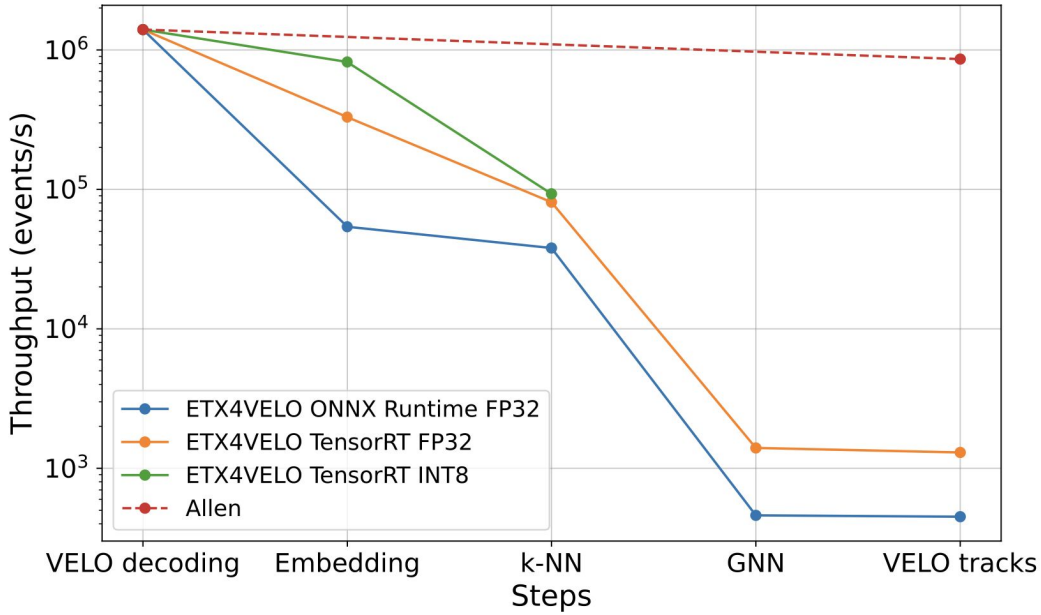
2407.12119



NVIDIA GeForce RTX 3090

Comparable or superior physics performance to Allen's velo track-finding algorithm, excellent electron reconstruction and low ghost rate

But throughput remains a challenge:
In the future considering: quantisation, NN optimisation

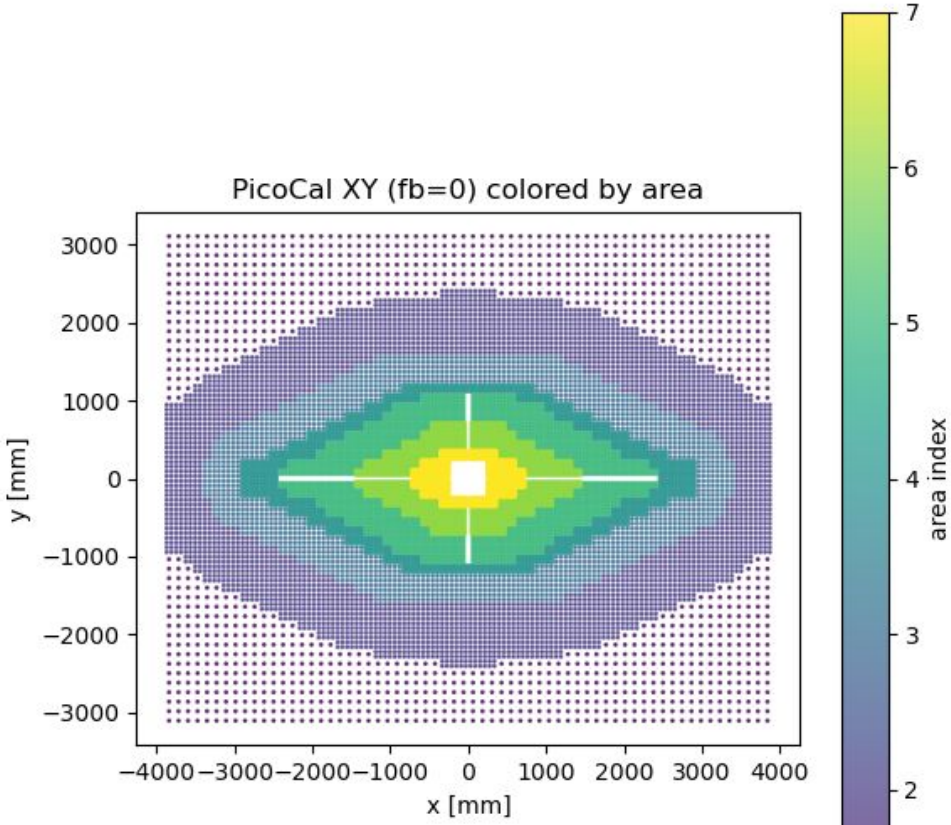


Upgrade II developments: Status

- Both GNN and GarNet models are developed and being studied **outside Allen**
- Integration in Allen needed as it is the official LHCb HLT1 application
- But it also provides utilities that are very useful for GPU R&D: multi-event scheduling (batching) & streaming, memory management, event and intra-event parallelisation etc

Status today:

- The classical approach is already implemented in Allen and will be used as a benchmark for other models
- Geometry and adequate data format of Run5 already in place
- Starting work on GarNet porting in Allen and inclusion of timing information



Conclusions

- Including ML techniques in the calo reconstruction is attractive for dealing with U2 conditions
- Both GNN and GarNet models have shown good performance in standalone frameworks.
- Timing information integration is still WIP

- But the question is:

  Can these models meet LHCb's real-time constraints, and how do they compare to the classical reconstruction in throughput?

  Allen can provide a platform to answer this question for the GPU solution

Backup

The ODISSEE project

- Online Data-Intensive Solutions for Science in the Exabytes Era
  - Huge (9.5M€) EU project — R&D for high-bandwidth science experiments
  - Both academic institutes and industry contributing (14 partners in total)
  - IN2P3 contributes to LHCb through the future calo and future HLT1

Some thoughts on the two approaches (GPU vs FPGA)

Clustering on FPGAs:

- ✓ Potentially can be performed at the same cards as for the readout - PCIe400?
- ✓ FPGAs more energy efficient than CPU/GPUs, potentially greener?
- ✓ Potential reduction of data to be transferred from detector to Event Builder
- – Harsh latency requirements
- – Full event information is not available

Clustering on GPUs

- ✓ Full event information available - no issues with cluster edges
- ✓ Less strict performance requirements
- – No data reduction between detector - Event builder
- – Could be more energy consuming