Finanziato dall'Unione europea NextGenerationEU

Ministero dell'Università e della Ricerca

Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA

ICSC Centro Nazionale di Ricerca in HPC, Big Data and Quantum Computing

4D vertexing with Graph Neural Networks

Jyoti Babbar (University of Trieste) for the CMS Collaboration

ICSC Spoke2 WP2 meeting– May 06, 2025

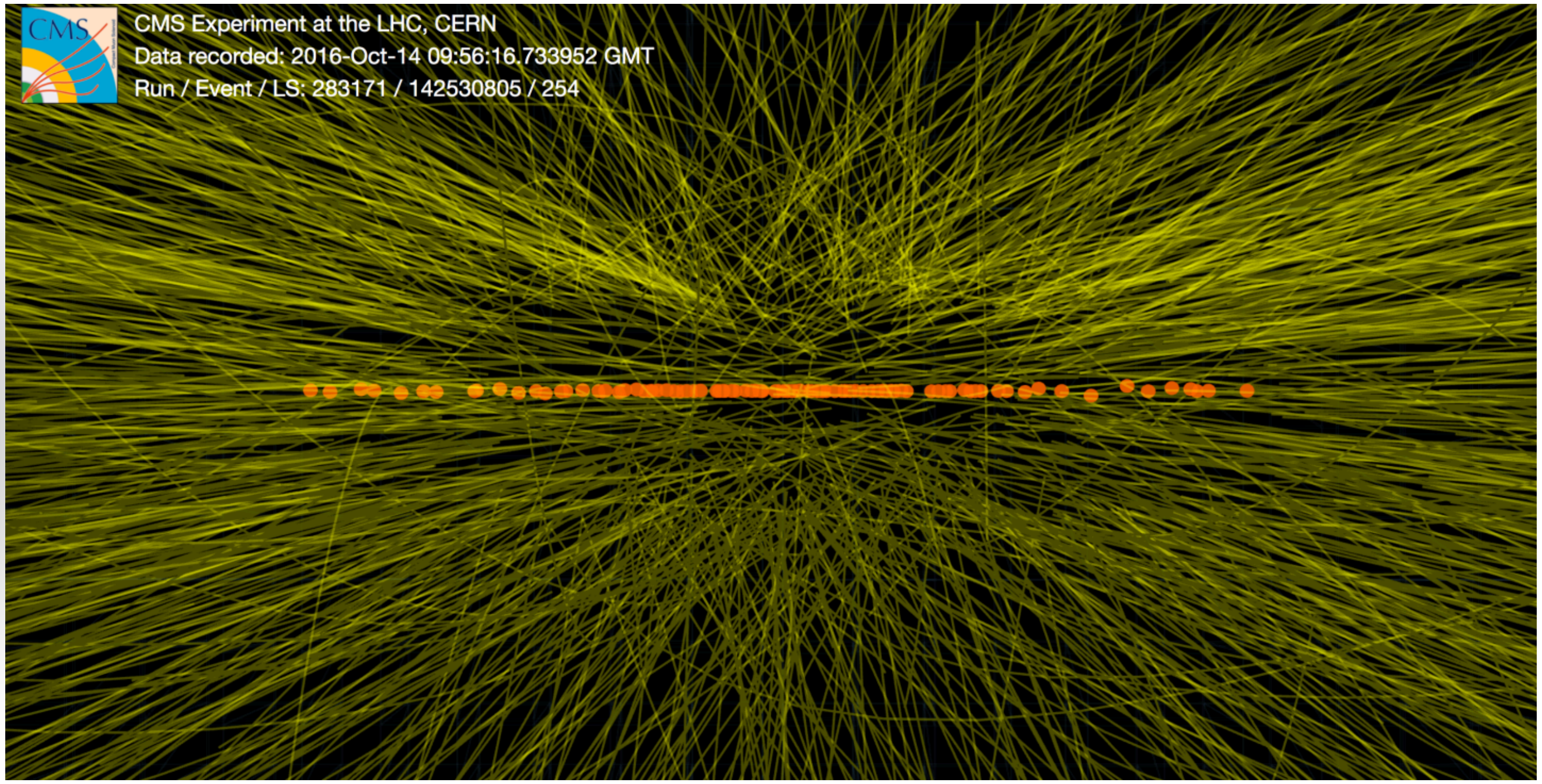
OUTLINE

- HL -LHC
- MIP Timing Detector
- 4D vertex reconstruction with Graph Neural Network
  - Edge Convolutional
  - GravNet
- Clustering Algorithms
- Conclusions

HL-LHC

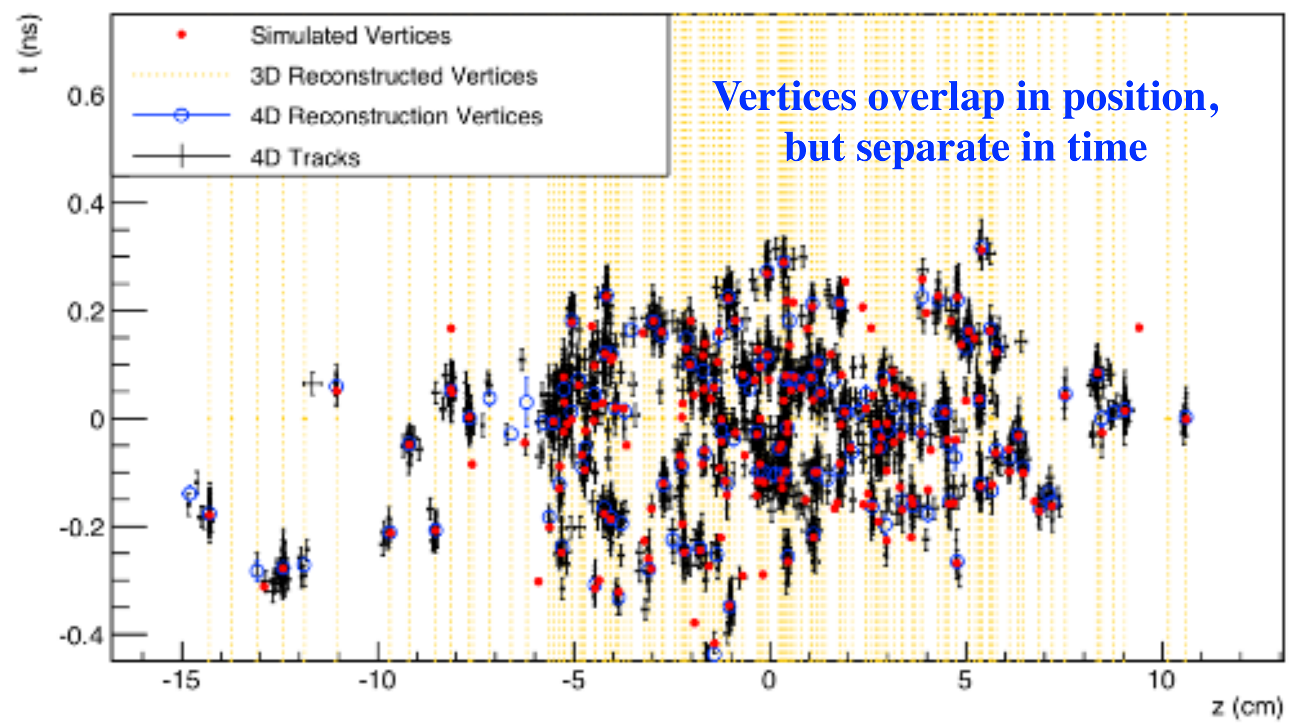
- High-luminosity LHC era (HL-LHC) starting in ~2030
  - x3-4 instantaneous luminosity
  - up to x5 pileup (PU) interactions
  - x10 integrated luminosity
- Crucial to isolate interaction of interest and mitigate effects of PU on object reconstruction
- Current global event reconstruction relies on track-vertex association in space

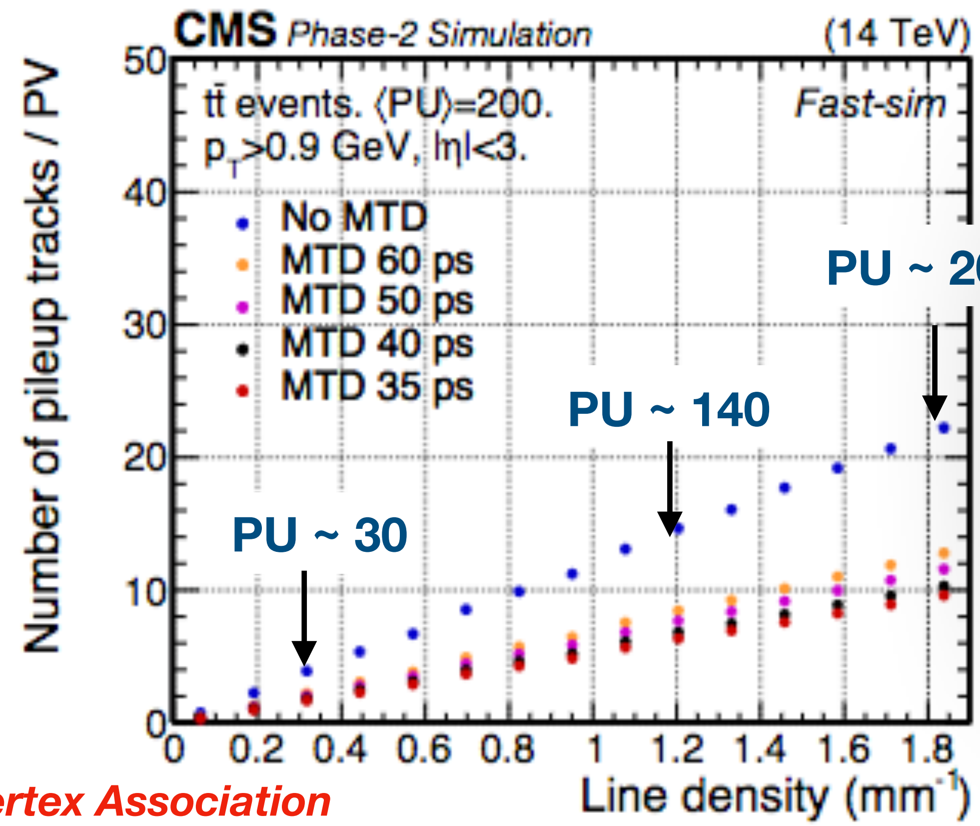
Challenge: keep current performance during HL-LHC phase



PRECISION TIMING AT CMS IN HL-LHC

- Minimum Ionizing Particle Timing Detector (MTD) proposed for the CMS experiment Phase2 upgrade
- MTD will provide timing information for MIPs with a 30-40 ps resolution ➜ smaller than the pp collision spread in time of 180-200 ps
- The vertical yellow lines indicate 3D-reconstructed (i.e. no use of timing information) vertices





Track-Vertex Association

4D vertex reconstruction — Use timing information to separate vertices that overlap in space

MIP TIMING DETECTOR

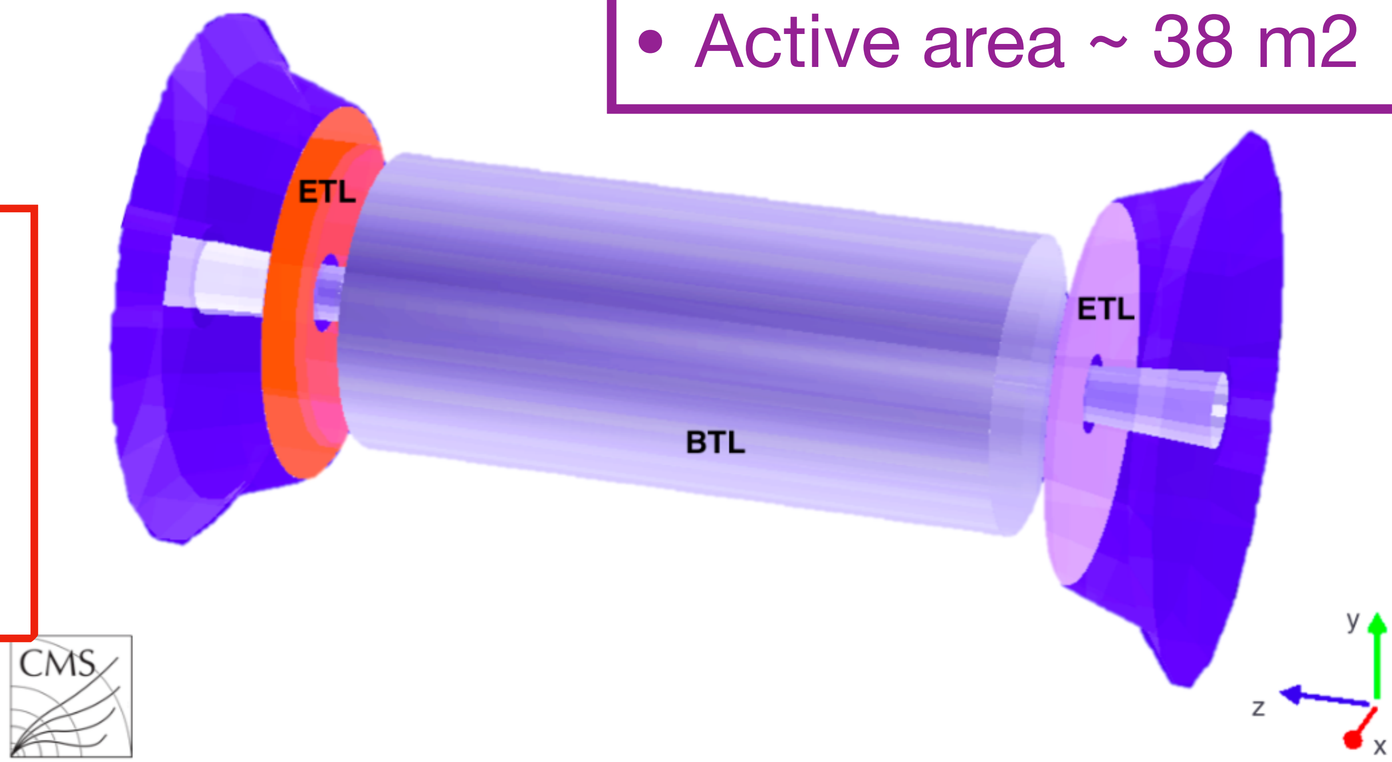
- Thin layer between tracker and calorimeters
- Almost hermetic ($|\eta|<3$)
- Different regions adopt different technologies, suited to the level of radiation dose:
  - Barrel Timing Layer (BTL)—arrays of LYSO crystal bars readout by SiPMs
  - Endcap Timing Layer (ETL)—Low Gain Avalanche Detector (LGAD) module

BTL

- LYSO Bars + SiPM readout
- $|\eta| < 1.45$ & $pT > 0.7$ GeV
- Active area ~ 38 m2

ETL

- Si with internal gain (LGADs)
- $1.6< |\eta| < 3.0$
- Active area ~ 14 m2

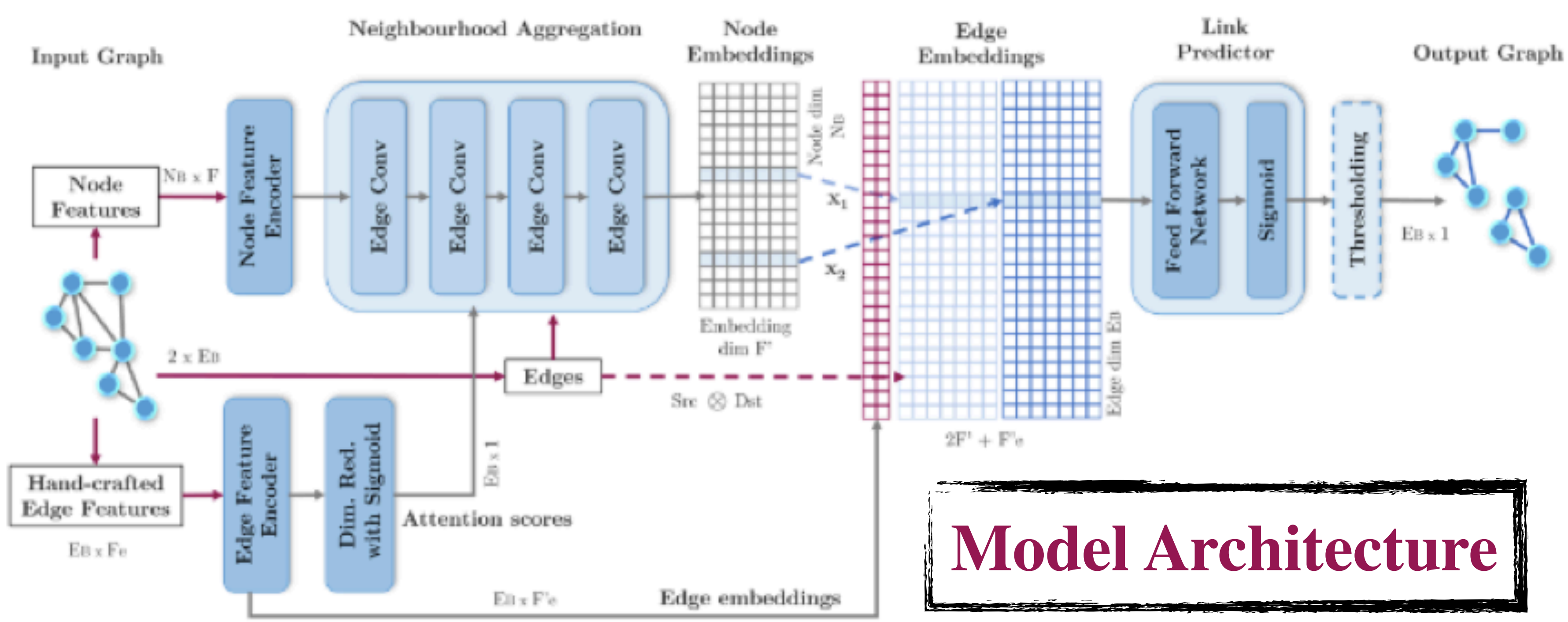


4D VERTEXING AND GNN

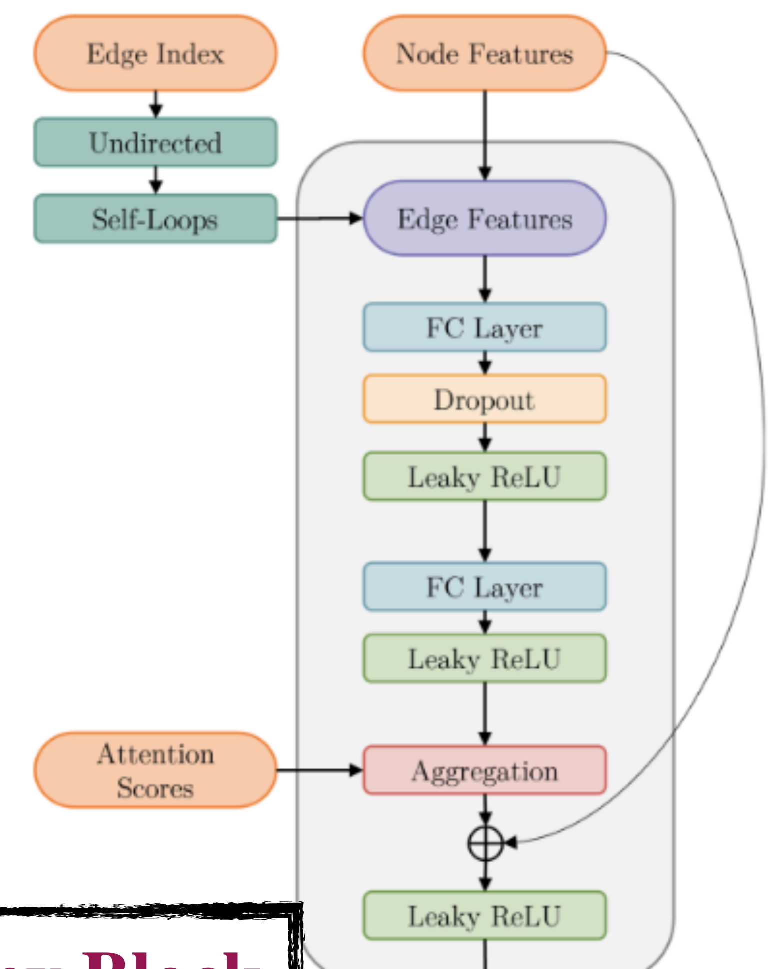
- New 4D vertexing algorithm shows significant improvement in separating close by vertices CMS-DP-2024-085
- However, incorporating Machine Learning architectures for track clustering particularly under high pileup conditions, could further improve performance.
- **Graph Neural Network (GNNs) are suited for track clustering and identifying track time/mass hypothesis**
- Two promising GNN architecture are studied :
  - GNN combined with Edge Convolutional **for message passing with a static pre-computed graph**
  - GravNet architecture with dynamic graph **where k-NN search is performed at each convolutional layer**
- For sanity checks of model architecture and directly comparing 3D vertexing with Deterministic Annealing (DA), we are just using the spatial information.

GNN + EDGE CONVOLUTIONAL

- The GNN+EdgeConv architecture is adapted from HGCAL trackster linking algorithm (HGCAL trackster), for nodes representing tracks and connection between them as edges.
- The model's output is given by an edge score between 0 and 1, with 1 indicating a good or true edge.



Model Architecture

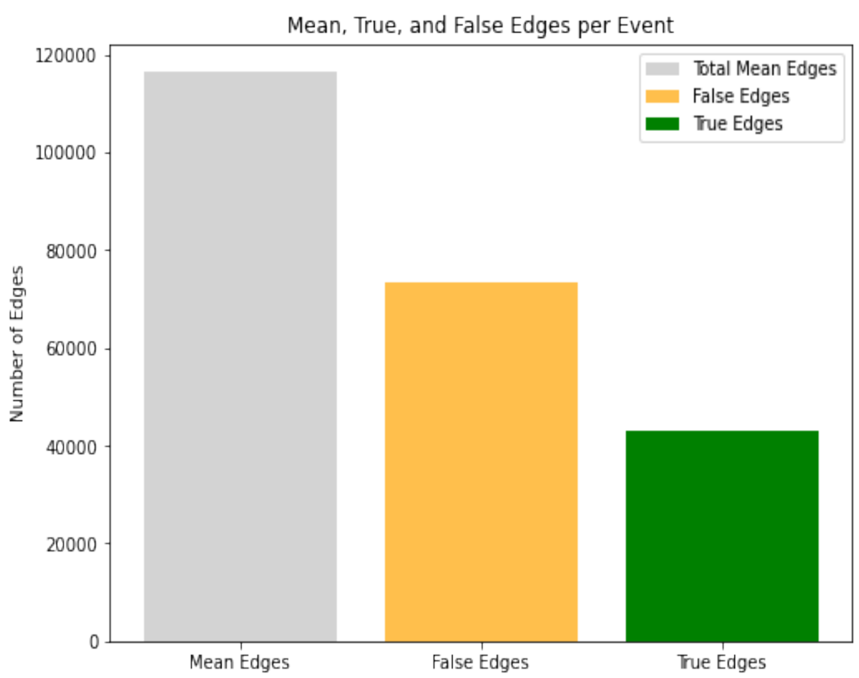


Edge Conv Block

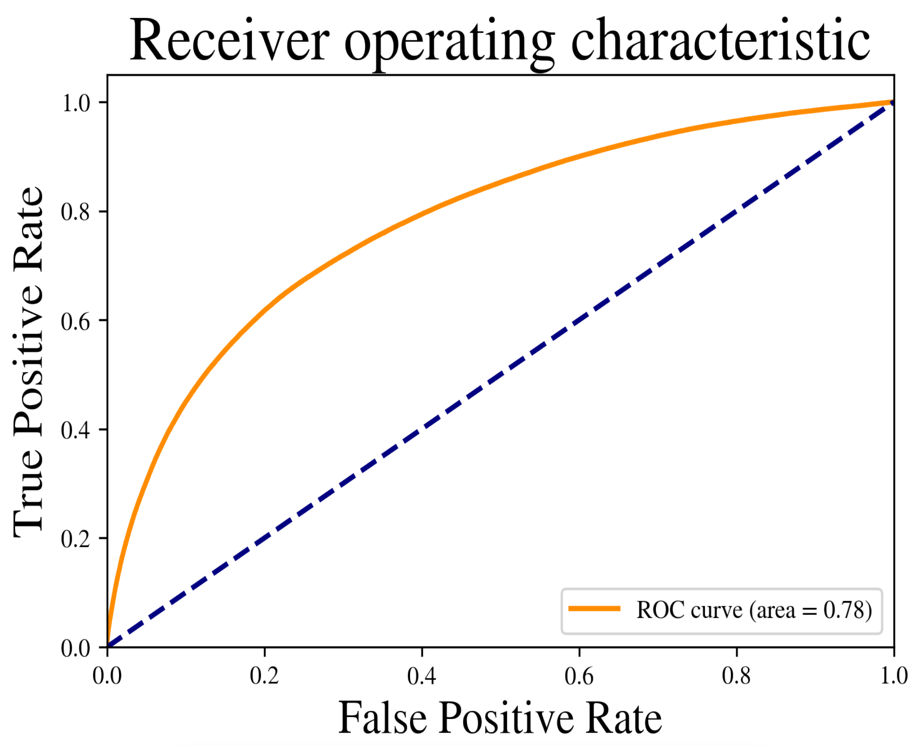
From J. Jaroslavceva's thesis

BUILDING GRAPH

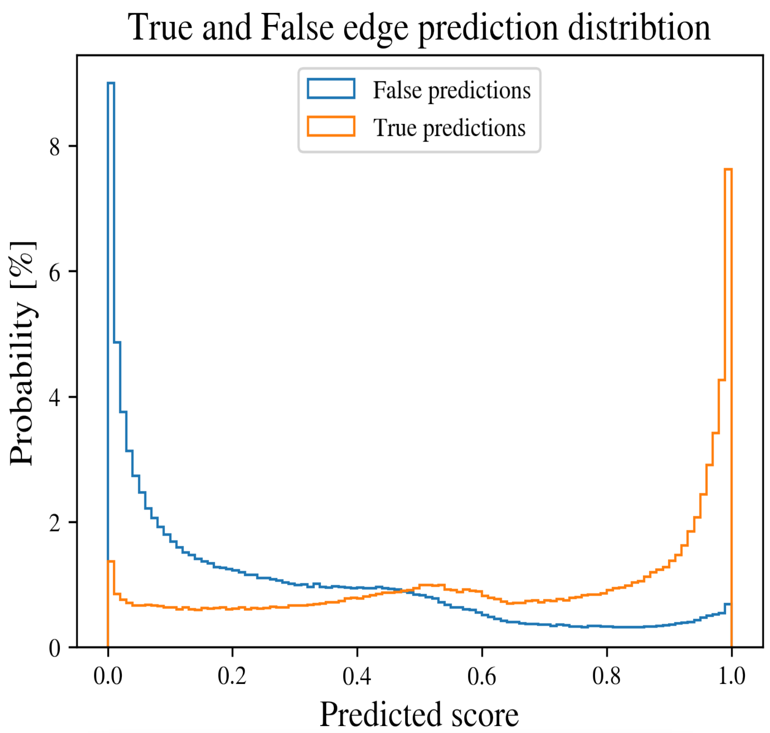
- Build graph with tracks as nodes
  - Node features : z, dz
  - Edge features : track-track z distance, and track -track dz significance
  - Possible edges between tracks with z-separation < 3 mm
- Defining True edges labels: that come from same vertex in an signal/PU event in BX == 0, same event ID and being the leading vertex of the event



True and False Edges Ratio



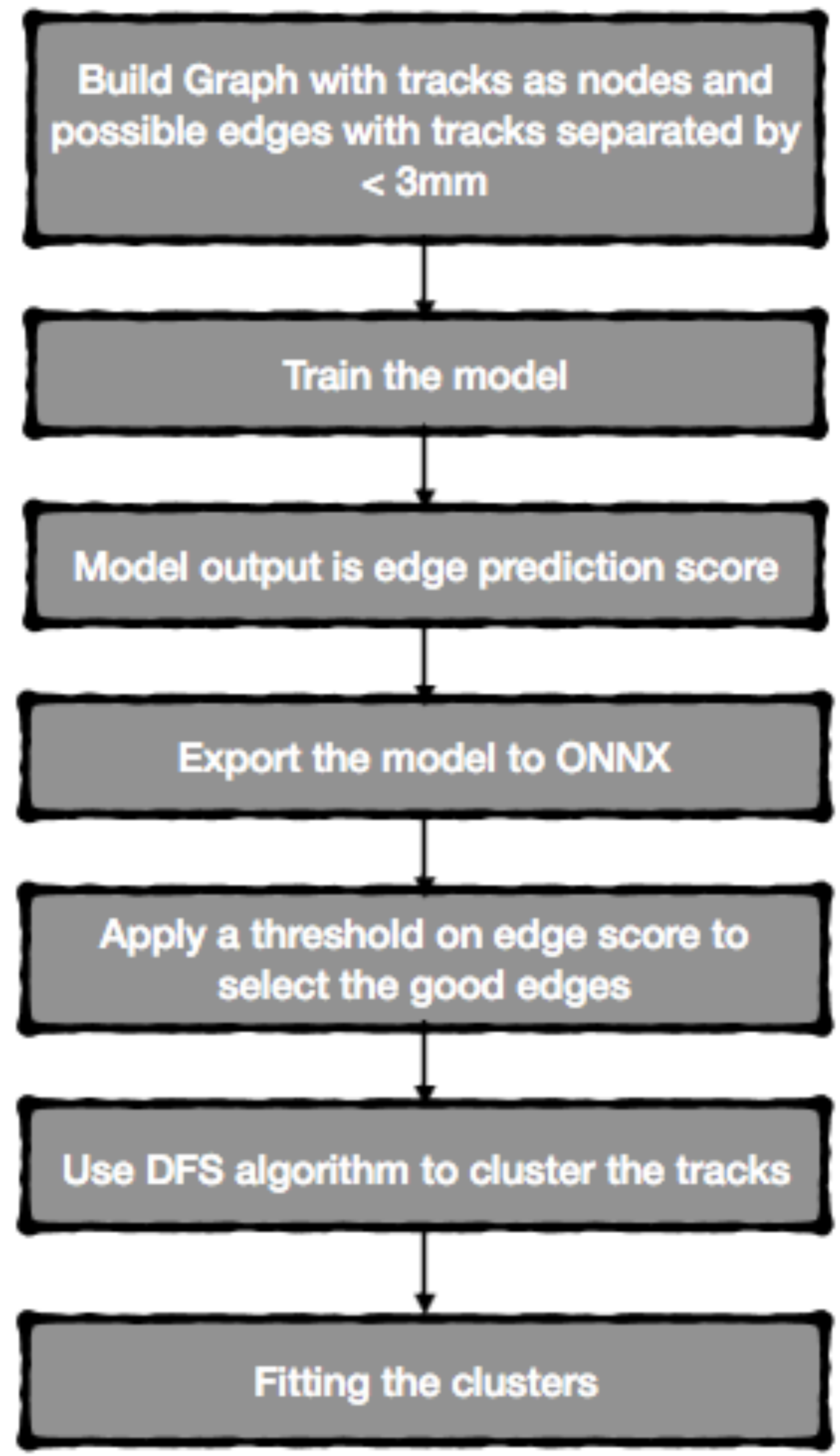
ROC Curve



Edge Curve

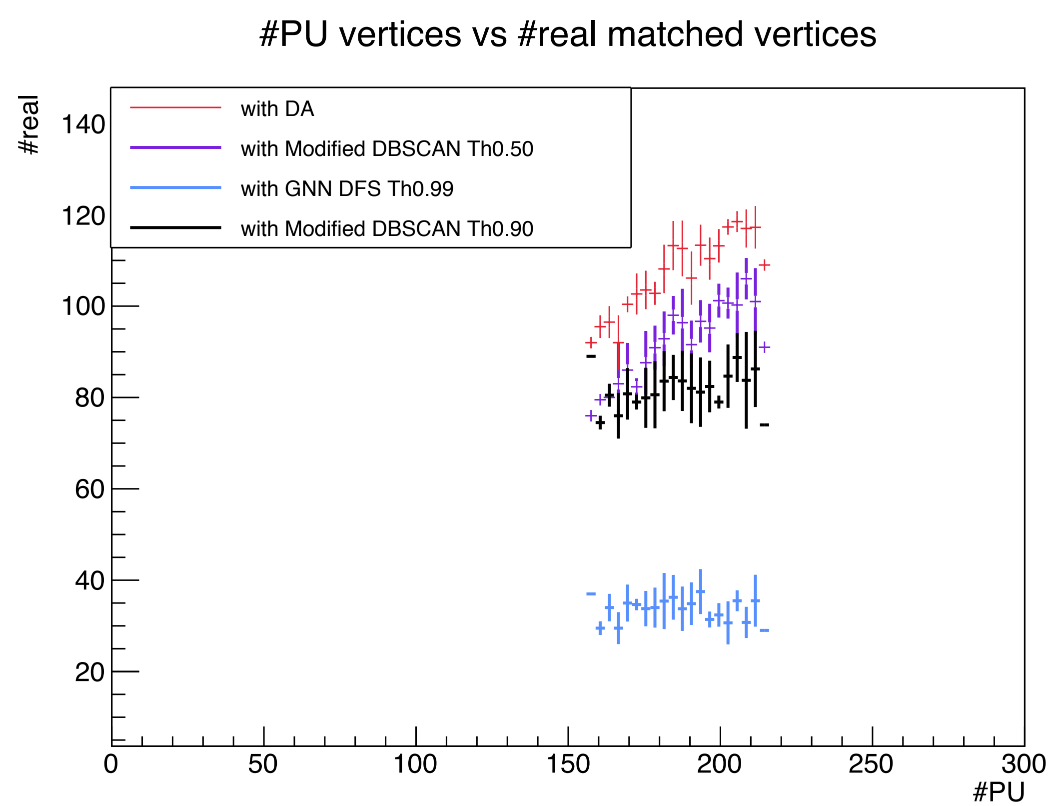
CLUSTERING

- The edge score for the tracks is calculated from the model to tracks and we decide a threshold to accept the good tracks
- Once the threshold is given, the algorithm cluster the tracks by finding the connected neighbors to a node, and which are above this threshold.
- We worked with two clustering algorithm to cluster the good connected tracks to form vertex candidate
  - **Depth First Search (DFS)**
  - **Density-based spatial clustering of applications with noise (DBSCAN)**. This model need two inputs provide two parameters ε (eps) and the minimum number of points required to form a dense region (minPts)

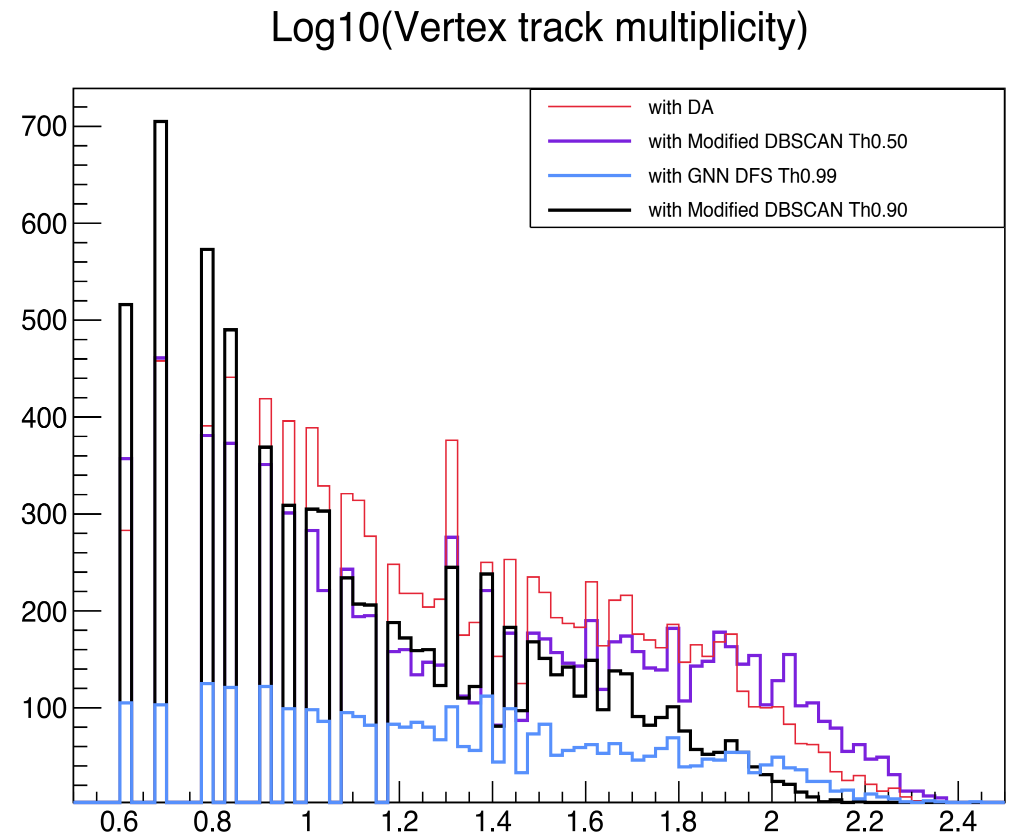


CLUSTERING

- We compare the results with Deterministic Annealing currently used by CMS for vertex reconstruction.
- DBSCAN and DFS algorithms are reconstructing less vertices than Deterministic Annealing



Real vertices



Vertex Track Multiplicity

Both algorithms are not an appropriate choice for vertex reconstruction —> significantly less number of total vertices. It was chaining the tracks and clustering far-apart tracks together.

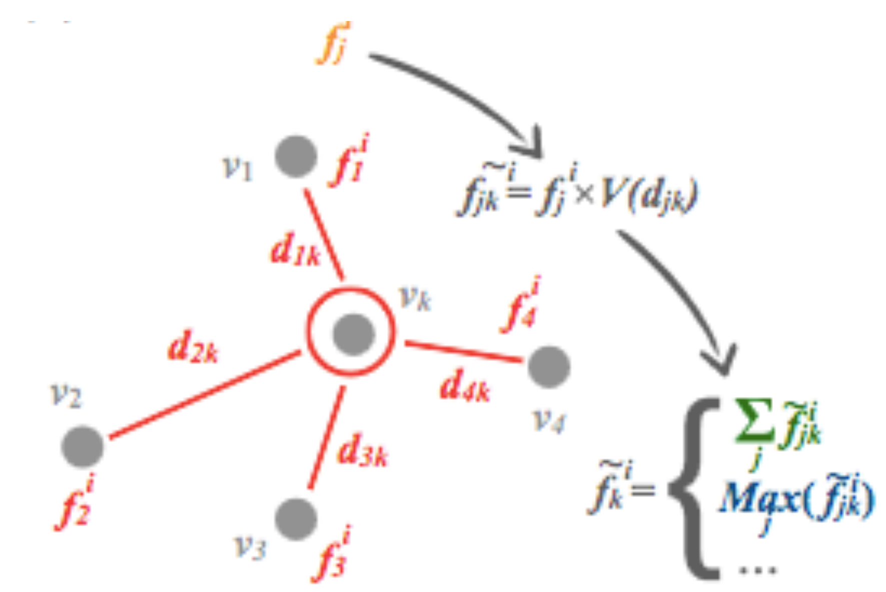
Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
ICSC Centro Nazionale di Ricerca in HPC, Big Data and Quantum Computing

GRAVNET

- GravNet model is the potential based model, i.e on the distance emphasizing the influence of close neighbors.

$$V_{ij} = exp(-d_{ij})$$

- The exponential decay ensures that nearby hits in the embedding space exert a larger influence.



Eur. Phys. J. C 79, no.7, 608 (2019)

Output of Model

- **Embeddings:**
  - Project tracks into a learned space where clustering can be performed.
  - Tracks from the same vertex are encouraged to be close in embedding space.
- **β :**
  - This indicates the "attractiveness" or importance of each track as a potential cluster center.
  - Tracks with higher values are more likely to be seeds for clusters (vertices).

LOSS FUNCTION & INFERENCE

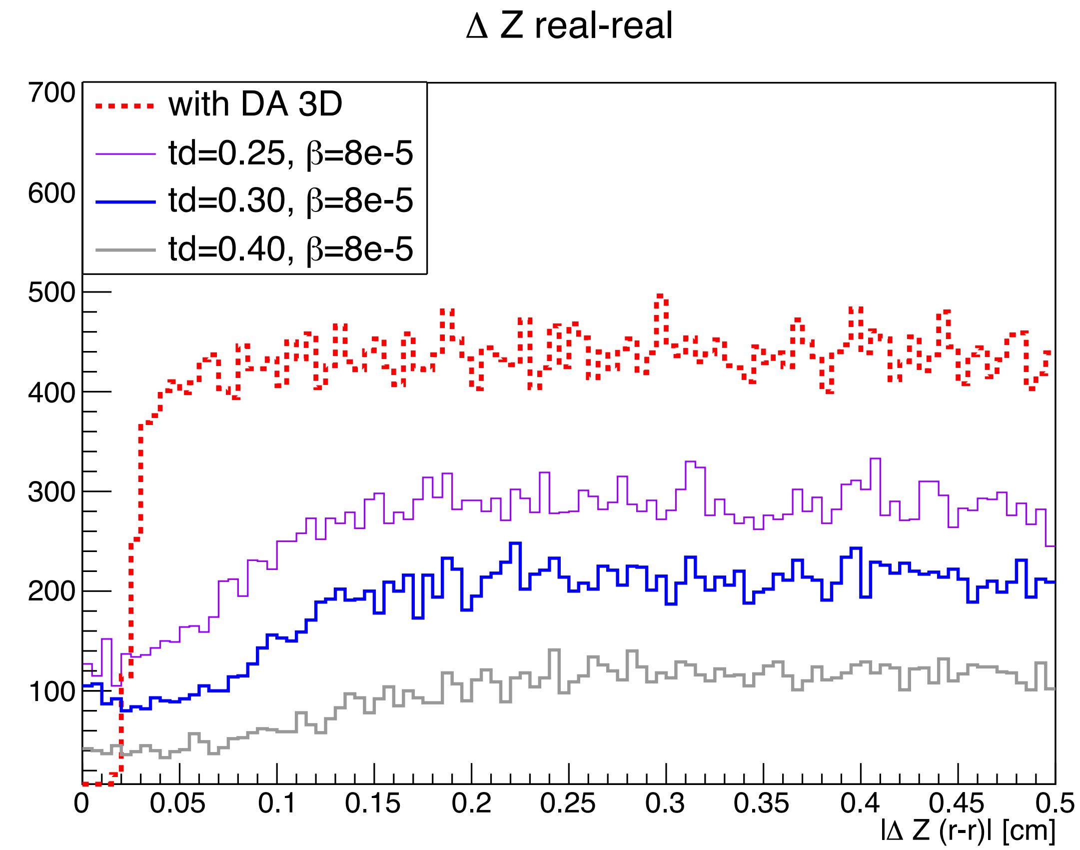
- Object Condensation method is used as the loss function following [2]. This method is independent of assumption on object-size, and can be generalized for graphs or point clouds.
- The total loss term in this method is composed of three main components : **Attractive Potential Loss Term , Repulsive Potential Loss term and β loss term (followed by [1] and [2])**
- The clustering of tracks is done based on β confidence output of the model [1]
  - Tracks are sorted in descending order of β value
  - Clustering starts with the highest **β** track, and within **td** distance parameter tracks are clustered and become a vertex candidate
  - Then, the process is repeated with the β ranking order.
- Highest β track is the condensation point (or seed) of a cluster

1. 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, no.7, 608 (2019), doi:10.1140/epjc/s10052-019-7113-9.

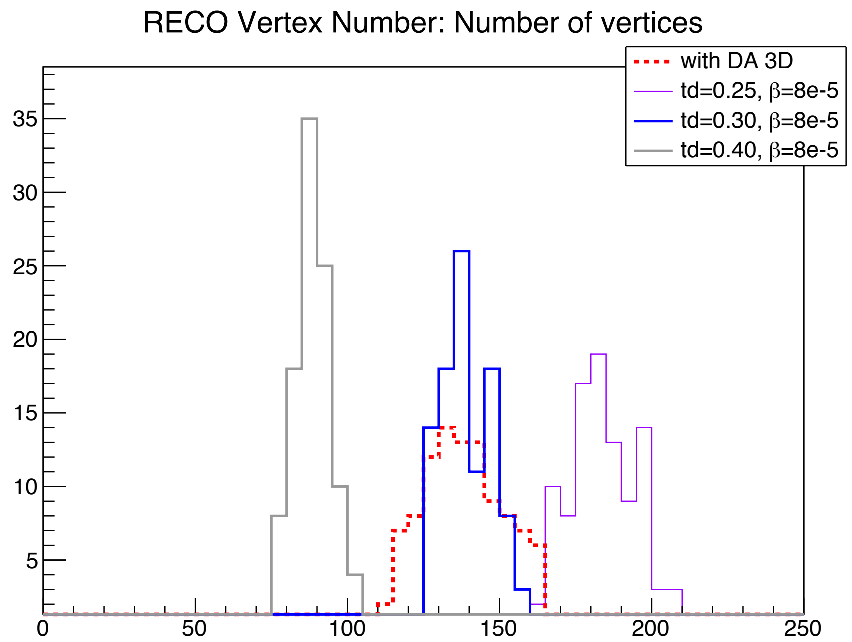
2. J. Kieseler, Object condensation: one-stage grid-free multi-object reconstruction in physics detectors, graph and image data, Eur. Phys. J. C 80, no.9, 886 (2020) doi:10.1140/epjc/s10052-020-08

RESULTS (WORK IN PROGRESS)

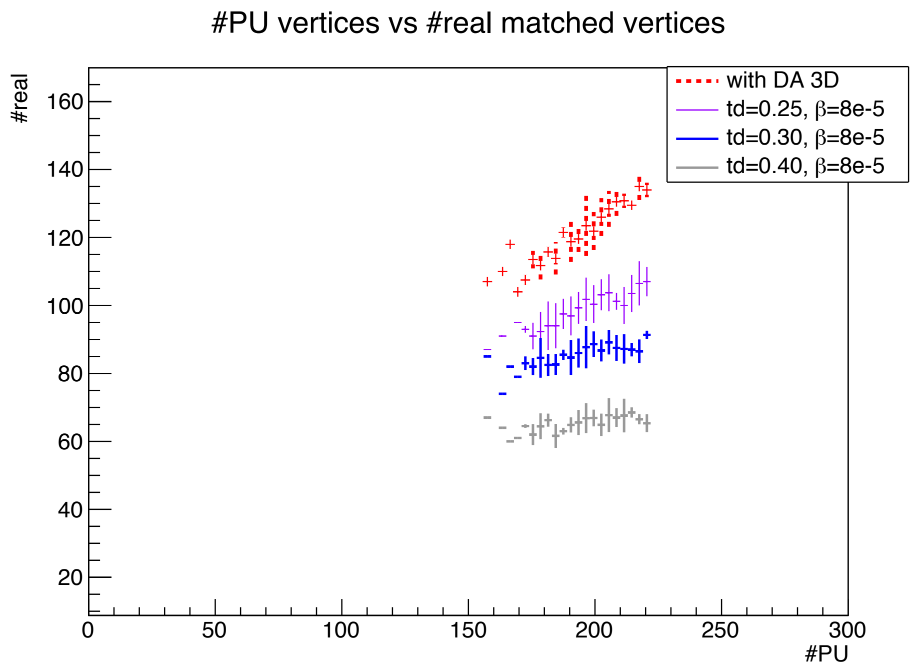
- Better separation of real vertices with smaller z difference.
- The number of reconstructed vertices as a function of the number of PU vertices for real and fake vertices
  - We get less real and more fakes than DA.
  - We are over-splitting the vertices.
- **We are currently refining the Loss Function to improve the model performance**



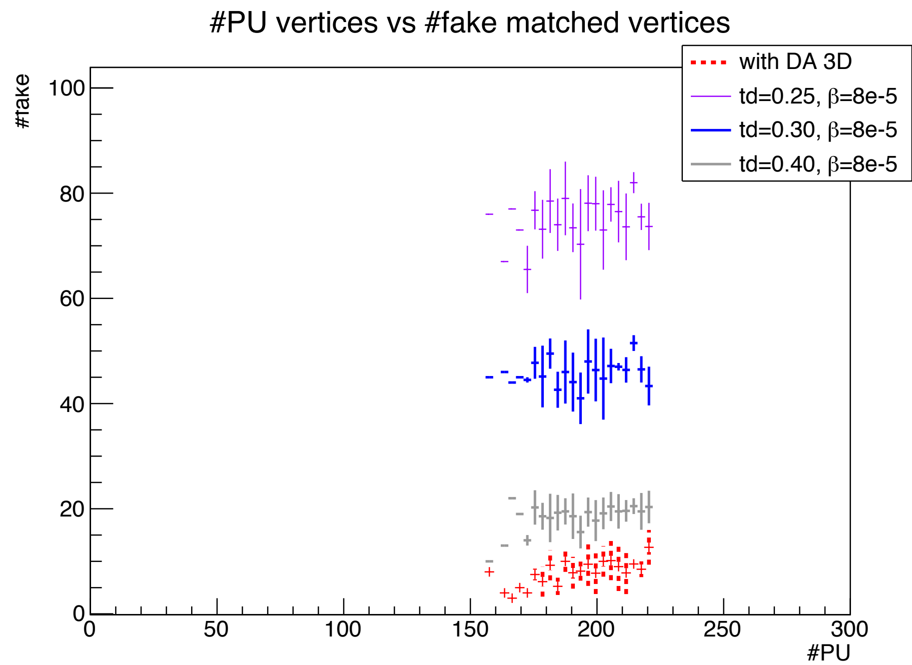
ΔZ separation



Total number of vertices



Real vertices

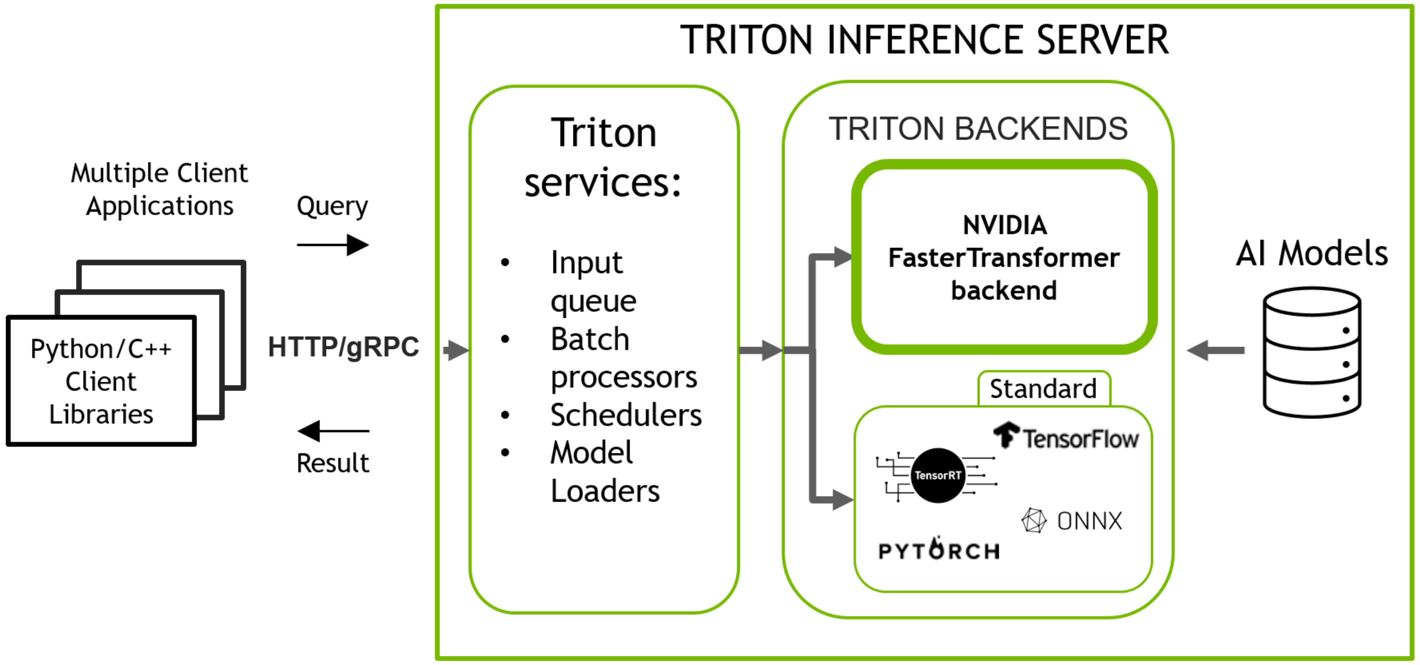


Fake vertices

COMPUTATIONAL STUDIES

- The inference of the model is consuming too much time and memory,
- We have started running the Inference on lxplus GPUs.
- We are testing our model with the SONIC framework (Services for Optimized Network Inference on Coprocessors) to do the inference of the model on Nvidia Triton Inference Server.

In SONIC —the ML model computations are offloaded to the NVIDIA co-processors



SUMMARY

- The timing information of MTD is important to mitigate the effects of PU in HL-LHC
- EdgeConv with DFS and DBSCAN is currently underperforming relative to expectations.
- **GravNet shows encouraging potential compared to EdgeConv, though model development and optimization are ongoing.**
- **Work is in progress to incorporate timing and additional input features into model training, along with loss function refinement for improved performance.**

Thanks !!!!!