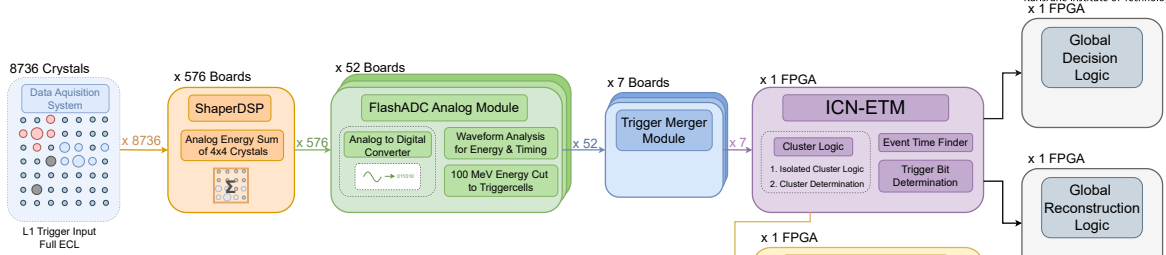
GNN-ETM Clustering Performance and Next Steps

50. B2GM TRG Parallel

Isabel Haide, Marc Neu, Torben Ferber, Valdrin Dajaku, Timo Justinger, Till Rädler | Monday 24th February, 2025

GNN for the ECL Trigger - Overview



- We have implemented the GNN ECL trigger algorithm on the GNN-ETM in parallel to the current ECL trigger algorithm on the ICN-ETM
- We would like to improve the algorithm and implementation for the next datataking period
- Marc has given a hardware overview, I will show a software overview and current to-dos and plans, and Torben will show the physics performance

Network Design of GNN ECL Trigger Algorithm

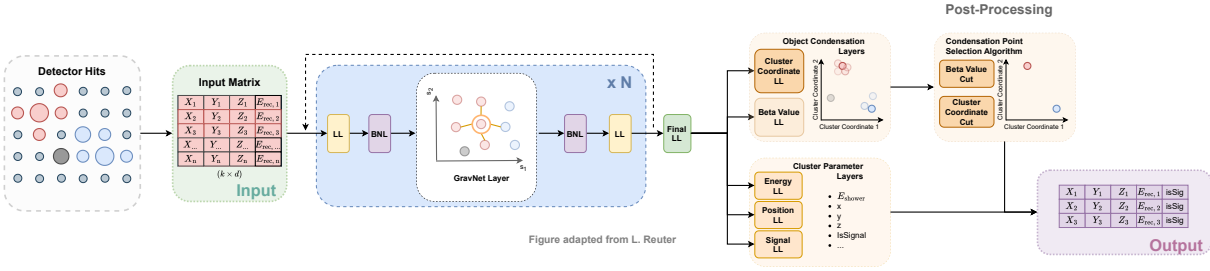


Figure adapted from L. Reuter

- Object Condensation (OC): One-shot algorithm for both detection and reconstruction of clusters (arXiv:2002.03605)
- Irregular geometry and varying input sizes in the ECL
  → **Graph Neural Networks (GNN)**
- GravNet layer dynamically builds graphs in a learned latent space and finds neighbours with a k-Nearest-Neighbour approach

- Implementation on FPGA requires max. 2 GravNet blocks and size reduction of linear layers
- Replacement of Euclidean distance for k-Nearest-Neighbour and Condensation Point Selection with Manhattan distance to reduce needed resources
- Using reduced fixed point precision for inputs and weights of layers to reduce number of calculations on FPGA

Training Inputs and Predicted Values

Karlsruhe Institute of Technology

Inputs:

- Inputs are all TCs > 100 MeV (currently applied energy cut for ECL Trigger operation) given by TSIM within trigger timing window (250 ns)
- Input features are reconstructed energy per TC, timing (relative to highest-energetic TC per event), x, y, z position of TC from lookup table

Dataset:

- We train on a technical MC dataset
- Dataset contains 1-6 photons with flat distribution in θ, ϕ and energy between 0.05 and 7 GeV
- We additionally simulate low energy photons to improve signal classifier performance
- We additionally increase the chances of overlapping clusters

Predicted Features:

- Training targets are offline ECL showers
- TCs have a 100 MeV energy cut: Only ECL showers that deposited the majority of energy in one TC are used as targets
- Predictions: Shower Energy, Shower Position (x,y,z), Signal/Background Classifier
- Shower energy and position are taken from the offline reconstruction
- A signal shower is defined as having > 30% MC energy deposition

Monitoring Metrics

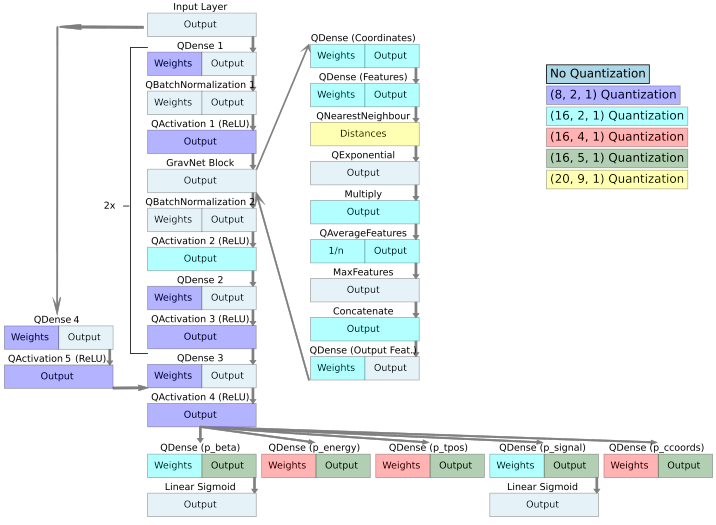
- We are monitoring efficiency, purity and energy and position resolution, as well as signal classifier performance
- Efficiency: N(corr. pred.) / N(true)
  - True clusters are all target ECL showers
  - Correctly predicted clusters are all GNN/TRG clusters matched to a target ECL shower
- Purity: N(corr. pred.) / N(all pred.)
- Resolution is currently calculated for all matched clusters

$$\mathrm{Res}(x) = P_{68\%}(|x - P_{50\%}(x)|)$$

(definition taken from the Belle II tracking paper arXiv:2003.12466)
with x = (E(pred) - E(true)) / E(true) for energy resolution
and x = (θ (pred) - θ (true)) for φ and θ resolution

- For signal/background classifier: Background rejection rate at 95% signal efficiency (comparison benchmark)

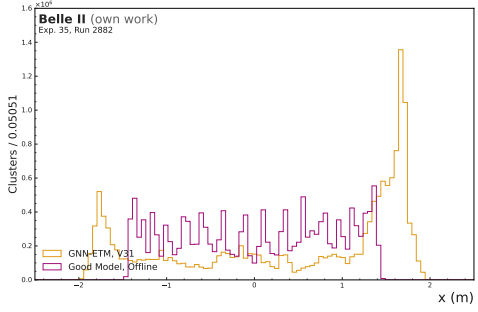
Model design and Quantization



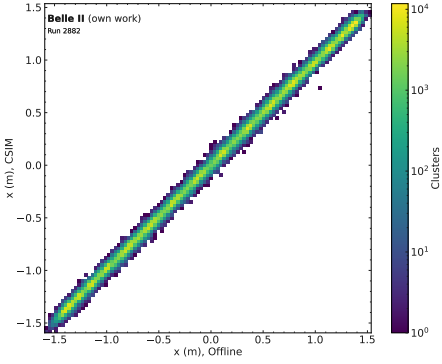
- Model has 11 linear layers in total (including GravNet layers) and 4700 trainable parameters
- Replacing Euclidean distance for k-Nearest-Neighbour algorithm and Condensation Point Selection algorithm with Manhattan distance
- All weights/biases/outputs are reduced in precision, going from 32bit floating point precision to 16 or 8bit fixed point precision
- Using QKeras for quantization-aware training, additionally pruning trainable parameters to 40% sparsity to decrease number of computations
- Exponential activation functions also had to be linearly approximated to decrease computations

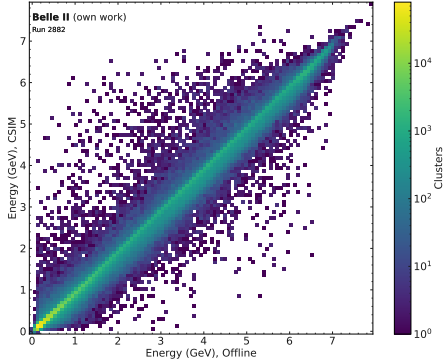
Current Status

- We have taken 0.19 fb^{-1} with the GNN-ETM with V31, but with bad model weights
- We can re-run the model inference offline with the GNN-ETM TC inputs
  - With the deployed model weights (orange) to confirm the behaviour of GNN-ETM (see Marc's presentation)
  - With better model weights, model name fine-gorge (purple), to analyze physics performance (see Torben's presentation)
- We are working on getting QKeras and C-Sim in agreement
- We are also working on calculating efficiencies etc. for the full dataset
  - We have to know very precisely which offline clusters are in the timing range of ICN-ETM and GNN-ETM to evaluate effects (work-in-progress)
  - We want to see especially the performance on close-by clusters and low energy resolutions



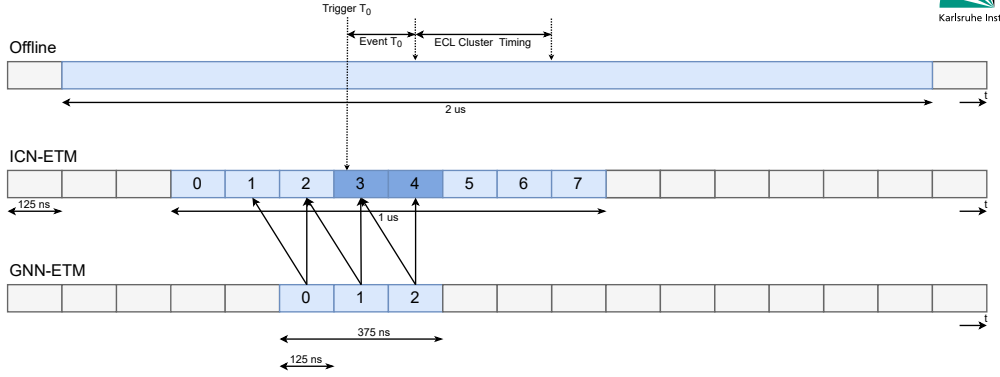
QKeras/C-Sim Agreement





- Agreement between QKeras (offline) and C-Sim is good for some features (example x-position prediction), but (too) bad for others (example energy prediction)
- We are trying to find the reason for disagreement (different rounding, different behaviour if quantization range is exceeded)
- For final models we can run inference with C-Sim to better model the hardware performance but for ongoing model development we would like to use QKeras

Choose correct offline Clusters



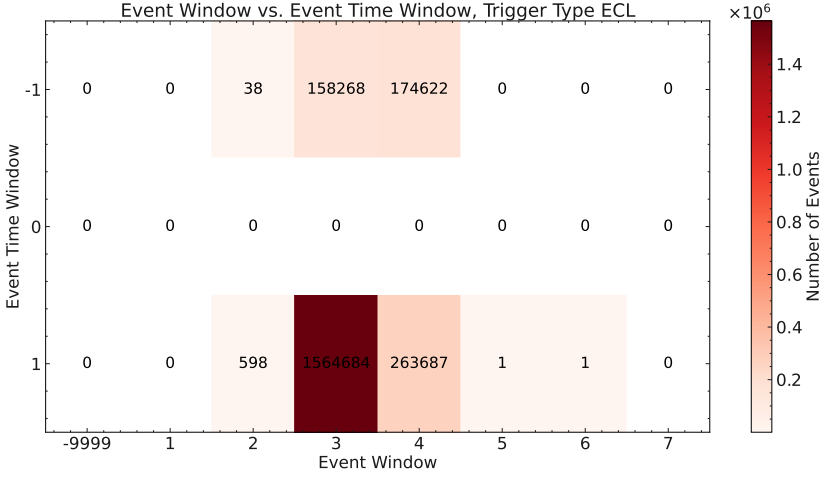
- ICN-ETM writes out 8 data windows with 125 ns and makes the trigger decision in two of those data windows (= trigger window)
- GNN-ETM writes out 3 data windows with 250 ns (125 ns overlap with the window before), with GNN-ETM window 2 corresponding to ICN-ETM windows 3+4
- EventT0 is the difference between the offline Event Timing and the TriggerT0 Timing
- TriggerT0 is the timing of the highest energetic TC in the ECL TRG Timing Window, if TRG Timing is given by ECL (99.6 % of Events in Run 2882)

Correct Timing Approximation

- We thought ICN-ETM windows 3+4 are the windows where the trigger bits are calculated in and where the timing is decided in
- Due to Unno-san's help, we realized this is not fully correct
- We can get two infos from `TRGECLUnpackerEvtStores`, the Event Window and the Event Timing Window
- The Event Window tells us the first window of two in which the trigger decision took place (so calculating the clusters and the trigger bits)
- The Event Timing Window is either 1, which means that the timing was taken from the same windows as the Event Window, or -1, which means that the timing was taken from the two windows shifted to the left
- E.g. Event Window = 3 means, that the trigger decision was made in 3+4, if Event Time Window = -1, then TriggerT0 was taken from windows 2+3

Event Window and Event Time Window for Type ECL

- Event Window and Event Time Window for Run 2882 with TRG Timing given by ECL
- Events with Event Window == 3 and Event Time Window == 1 are what we expected
- If Event Window ≥ 4, GNN-ETM has either only one or no data windows overlap with the actual trigger decision



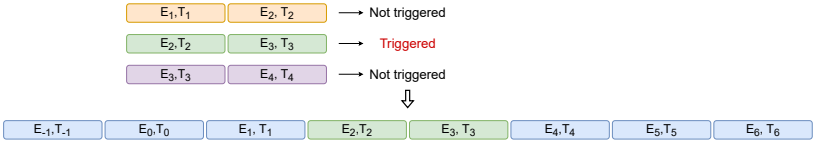
Open Question: Event Window/Event Time Window

Karlsruhe Institute of Technology

- How is decided which data windows are the Event Window?
- My current understanding:
  - The ICN-ETM calculates clusters and trigger bits for every 256 ns
  - GDL does not want adjacent triggers so we decide with an algorithm which trigger window should be used if two adjacent trigger windows would trigger
  - The trigger window which makes the trigger decision is always defined as data windows 3+4 and we read out the 3 windows before and after
- Due to different Event Windows and Event Time Windows, something different has to happen at one step?

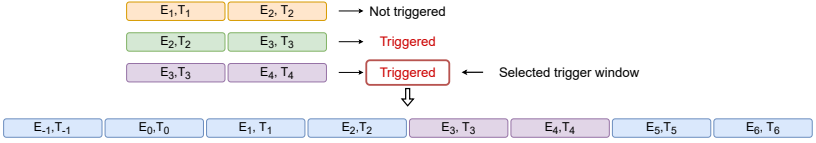
Open Question: Event Window/Event Time Window

Case 1:



ECL TRG Readout

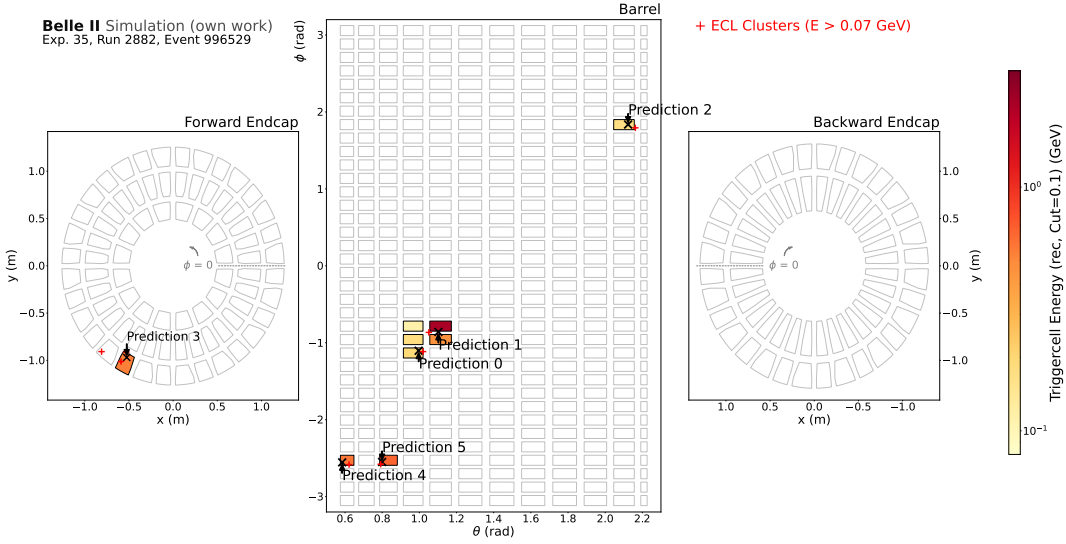
Case 2:



ECL TRG Readout ?

- Case 1 should result in Event Window == 3 and Event Time Window == 1, anytime only one window and no adjacent windows are triggered
- Case 2 results in Event Window == 4?

Event Display - GNN TCs with good Prediction



Linear Approximation of Activation Function

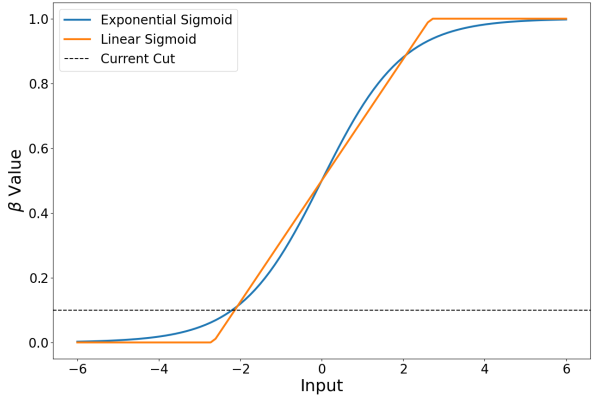
- Some prediction values, such as the β value for the condensation point selection algorithm, have to be constrained between 0 and 1.
- This is usually done via the sigmoid function:

$$f(x) = (1 + e^{-x})^{-1}$$

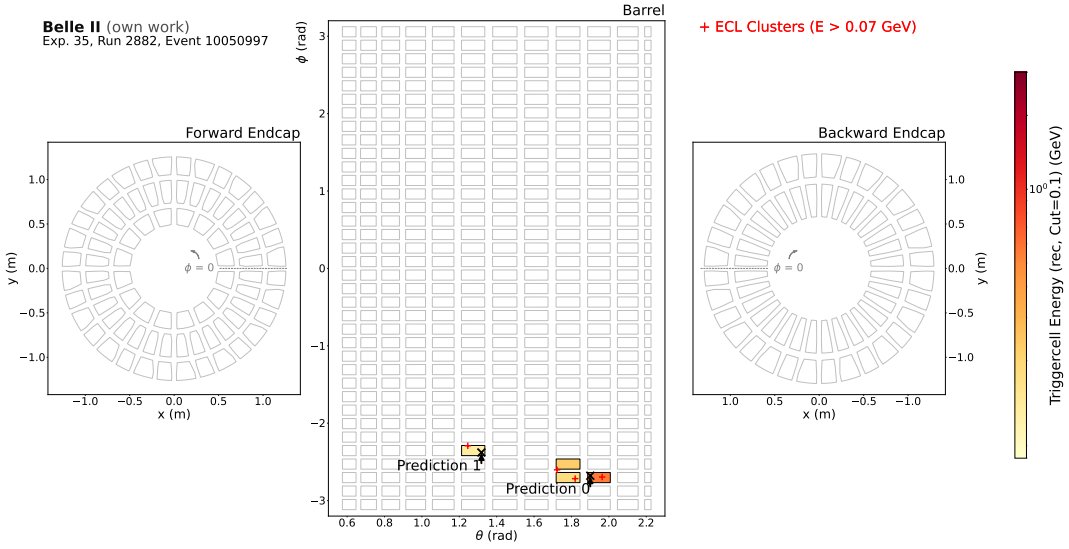
- In QKeras and in the hardware we approximate this by a linear sigmoid:

$$f(x) = \text{clip}(0.1875x + 0.5, 0.0, 1.0)$$

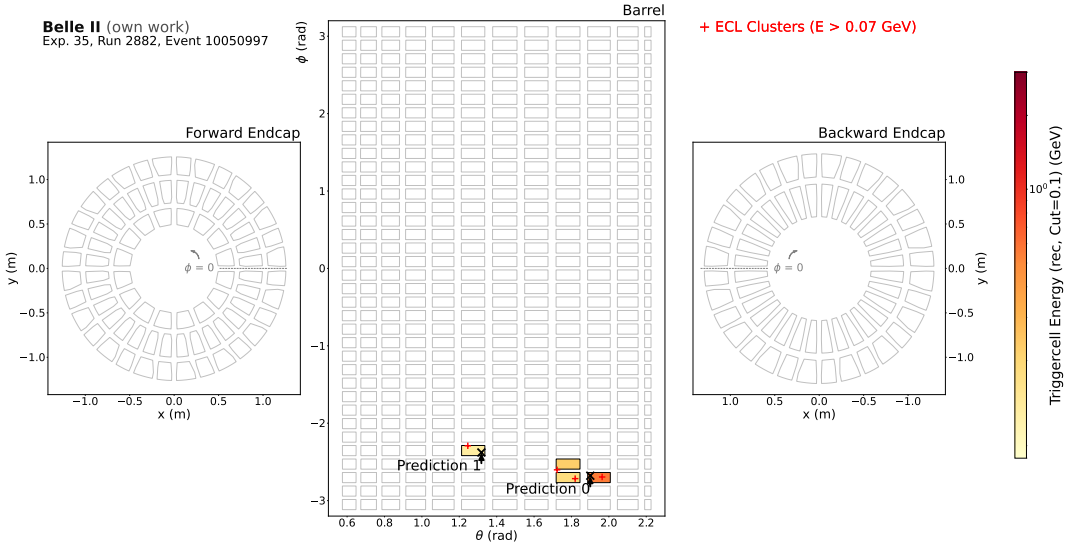
- We currently set the cut for β to 0.1, which means no point that has a predicted β value below 0.1 can be a cluster candidate
- This value is usually tunable to change the performance of the network, but significantly less with the linear sigmoid



Event Display - Prediction with Linear Sigmoid, β Cut 0.1

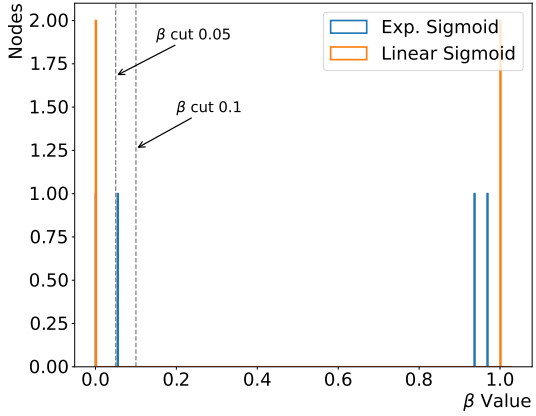


Event Display - Prediction with Exp. Sigmoid, β Cut 0.1

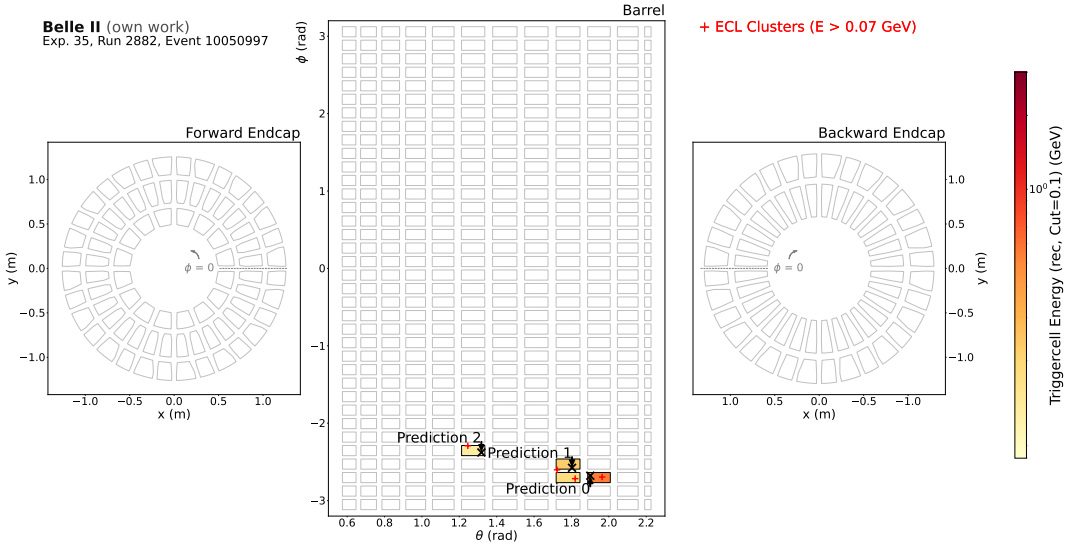


Comparison of β Values

- If we keep the same β cut of 0.1, the predictions do not change because this is the overlap point between linear and exponential sigmoid
- With the exponential sigmoid we can tune the β cut to improve predictions on overlapping clusters for example
- I'm showing one example event here, I am currently checking the full impact
- By adapting the linear approximation, we can maybe keep the working point adjustable



Event Display - Prediction with Exp. Sigmoid, β Cut 0.05



Next Steps and Tasks

Status

- We checked comparison between QKeras and C-Sim
- We are running the inference on the data GNN TCs to check the performance of the network with good model weights
- We are analysing the performance of the GNN-ETM in comparison with the ICN-ETM on selected physics cases (see Torben's talk)

Current ToDos

- Check metrics (efficiency, purity, resolutions) on cluster level and performance of close-together clusters
- Check impact of linear modelling of activation functions

Open Tasks

1.) Improve agreement between C-SIM and QKeras modelling

2.) Improve network performance for operation
   - Improve low energy resolution
   - Tune quantization values and pruning sparsity
   - Test algorithmic changes to decrease latency

   $\rightarrow$ New master students (F. Baptist, T. Lobmaier) at ETP

3.) Create merge request for GNN-ETM unpacker and dataobjects

4.) Write DQM Plots for GNN-ETM for next datataking period

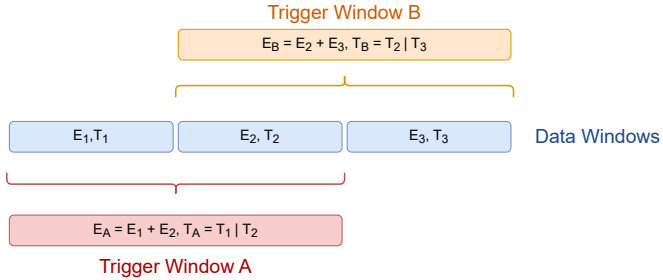
5.) Include GNN-ETM in TSIM
   - We won't be able to do this before the feature freeze for release-10

6.) Write documentation and make it available

Backup

Back-Up

Decision for Event Window



- As Unno-san explained to us, GDL does not want adjacent triggers, so a comparison logic decides which trigger window to use
- Possibility 1: If $E_A \geq E_B$, then trigger window A is chosen, otherwise window B
- Possibility 2: If If $E_1 \geq E_2$, then trigger window A. If $E_2 > E_1$, then we check if $E_A \geq E_B$, if yes trigger window A, else trigger window B.