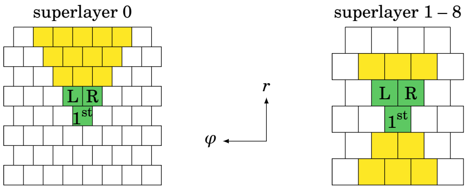
Optimized Inputs for the Neuro-Track-Trigger and Principles for a Displaced Vertex-Trigger

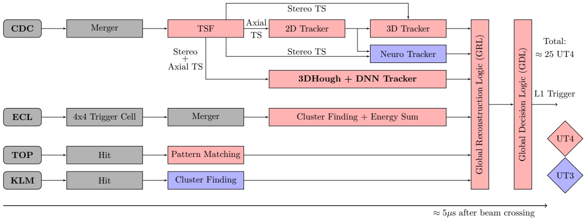
Timo Forsthofer, Christian Kiesling, Kai Unger, Simon Hiesl

May 3, 2024

Extended Input

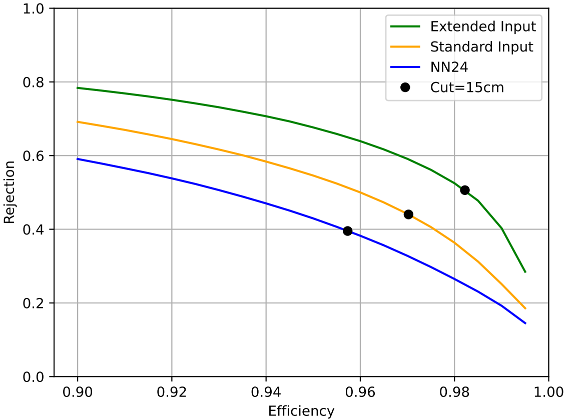
- New hardware after LS1 makes use of all track segment wires possible
- Drift times of wires added to current input, -1 if wire is missed
- No left/right information for non-priority wires
- Wire pattern always consistent, so information about hitpattern can be extracted

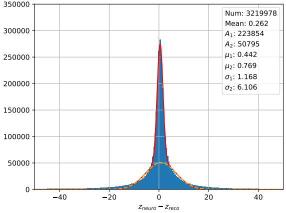




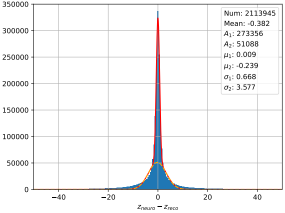
Extended Input

- Networks with 4x100 hidden nodes with the standard 27 input nodes (DNN) and the new 126 input nodes (XI) compared to currently implemented network (NN24)
- For z-Cut at 15cm, Rejection Rates are 39.5% (NN24), 44.0% (DNN) and 50.6% (XI), with Efficiencies at 95.7% (NN24), 97.0% (DNN) and 98.2% (XI)





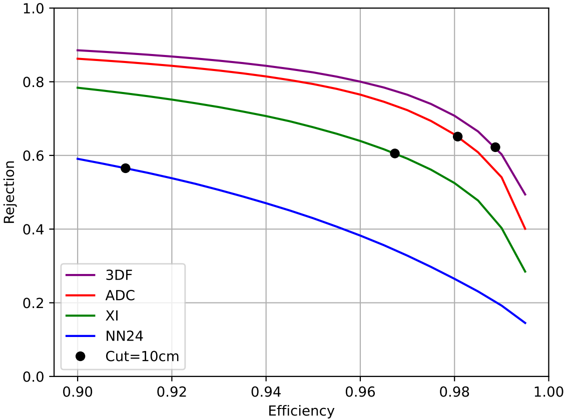
DNN



XI

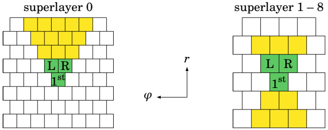
Further Improvements

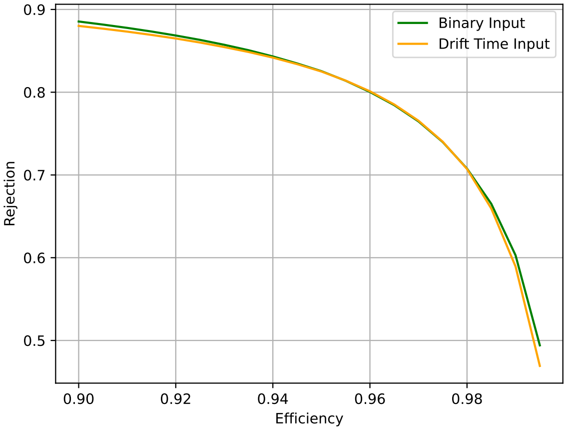
- ADC-cut attenuates background and makes signal tracks easier to identify (ADC)
- 3DFinder provides more accurate parameters and already preselects tracks (3DF)
- Network was reduced to 4x60 nodes with no negative effect on performance
- For z-cut at 10cm, 3DF gives lower rejection rate (62.2% instead of 65.1%), as the 3DFinder already filters out background that is then missing in the statistics



Optimized Extended Input

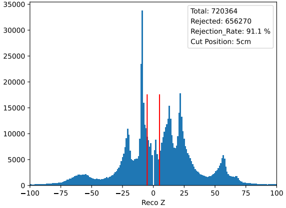
- Drift times in non-priority wires not meaningful without left/right information
- $\rightarrow$ Simply pass 1 if ADC-count above threshold, 0 else
- Easier to implement into hardware and makes bigger first hidden layers possible
- No negative effect on performance



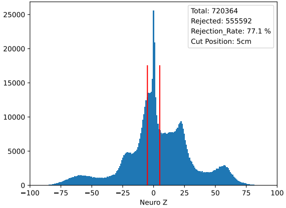


Classification Node

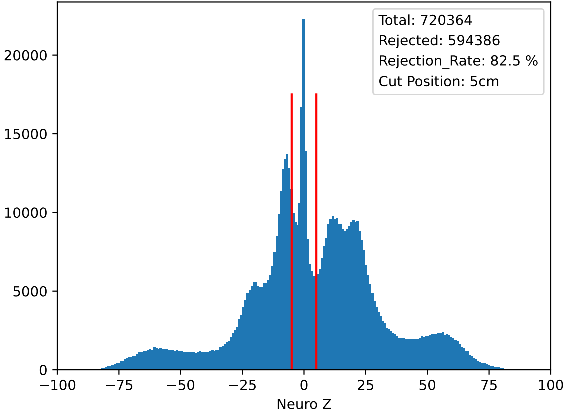
- Third output node is trained to 1 for $|z_{reco}| < 1cm$ (signal), 0 for $|z_{reco}| > 1cm$ (background)
- No longer good loss for prediction of background at IP $\rightarrow$ background prediction more accurate and less Feed-Down



Reco for $|z_{reco}| > 1cm$



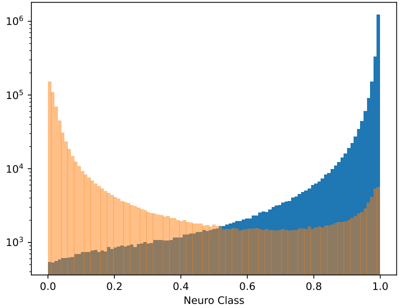
Previous Best Network

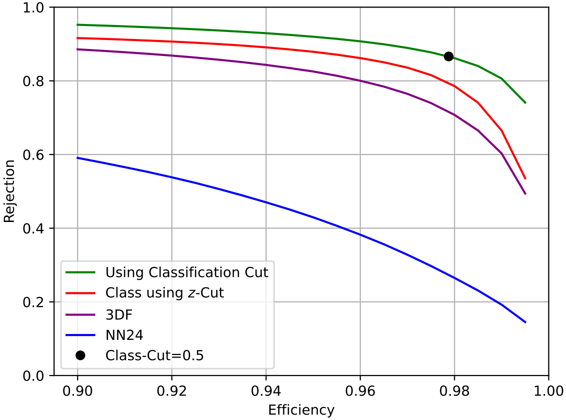


Background Prediction with Classification Node

Classification Node

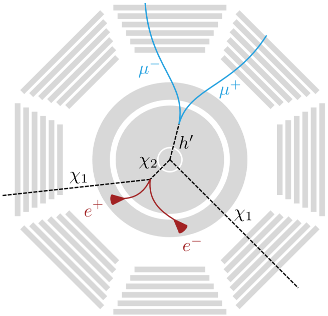
- Newly trained network (Class) already better with z-cut
- Classification node quite neatly separates signal and background
- Wide range of different cuts viable
- With cut at 0.5, Rejection Rate of 86.6% with Efficiency of 97.9% possible

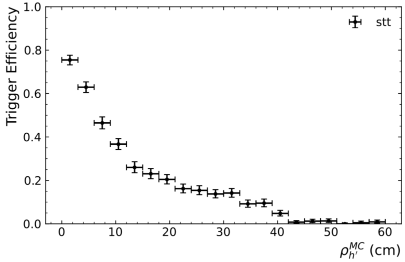


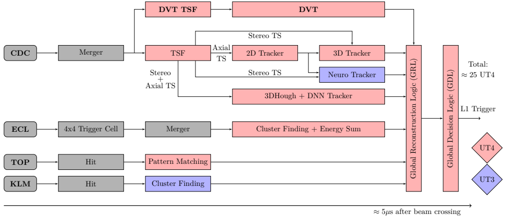


Displaced Vertex Trigger: Motivation (Master Thesis Elia Schmidt)

- Current CDC-Trigger is restricted to particles from the vertex
- Efficiency rapidly decreases with vertex displacement
- $\rightarrow$ Need Displaced Vertex Trigger in order to detect rare events with feebly interacting massive particles (for example dark matter)



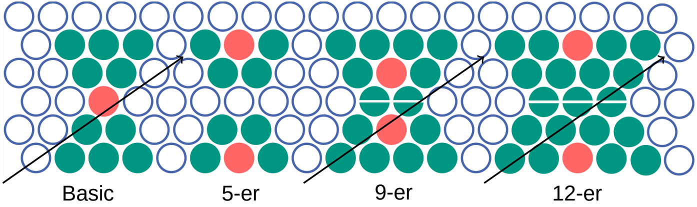


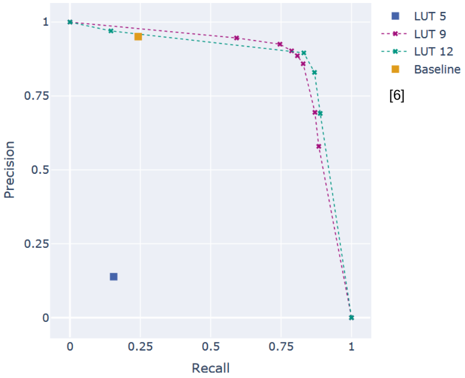


MC model: https://doi.org/10.1007/JHEP04(2021)146

Preprocessing

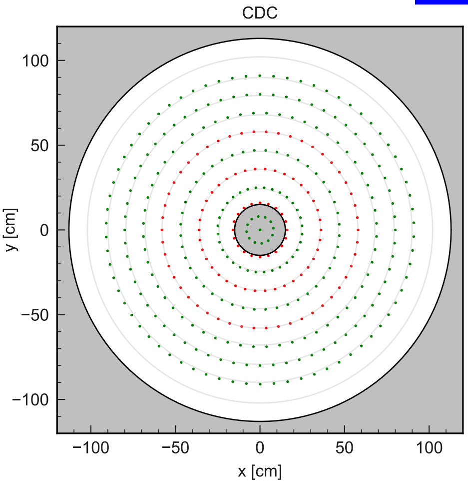
- Traditional Track Segment Finder (TSF) bad at detecting shallow tracks $\rightarrow$ new TSF by Mark Neu
- Look Up Table stores patterns indicative of signal tracks
- ADC-cut applied after TSF to clean Hough matrix





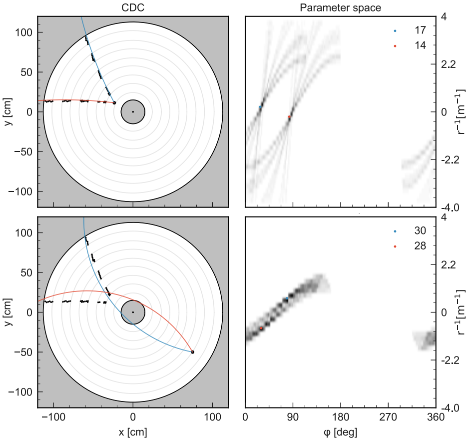
MacroCells

- 100 MacroCells throughout the CDC
- Parallel Hough Transformation for every vertex assumption
- Goal: Find two tracks with opposite charge originating at the same vertex



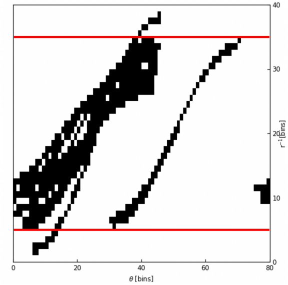
Selection of Best Vertices

- Wrong vertex assumptions often give high Hough peaks, but shape is very different
- Addition of Hough matrices of all hits computationally very expensive
- $\rightarrow$ Different approach needed

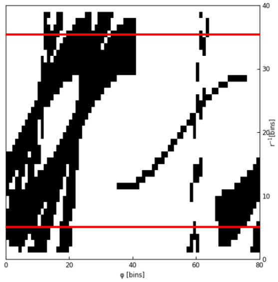


Selection of Best Vertices

- Hough matrix reduced to bitmap → very fast calculation
- Score is number of hits with small radii divided by number of hits with large radii
- Five vertex assumptions with highest scores are selected and further processed



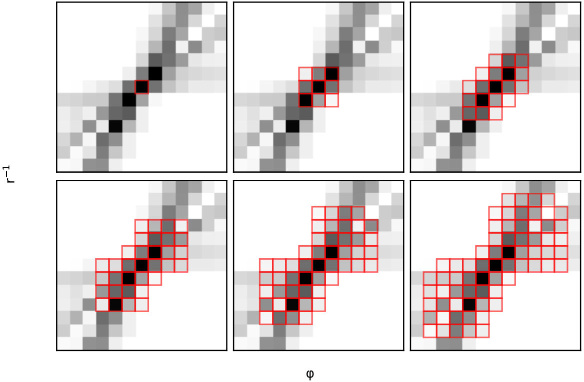
Wrong vertex assumption



Correct vertex assumption

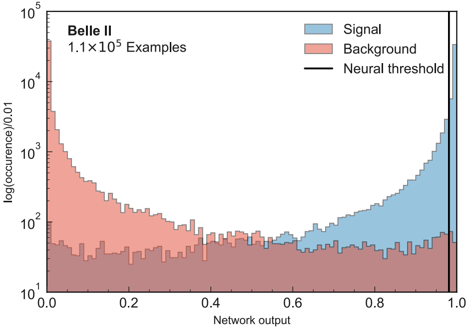
Neural Clustering

- Best vertex assumptions from the previous step analized more thoroughly
- Start at peak and add direct neighbors to cluster
- At most five iterations to ensure deterministic computing time
- Ten cluster parameters, such as track curvature and 2D opening angle, are calculated and passed to neural network



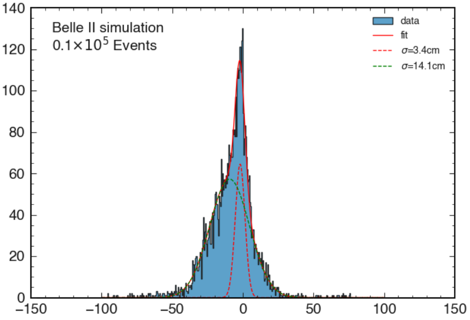
Neural Network

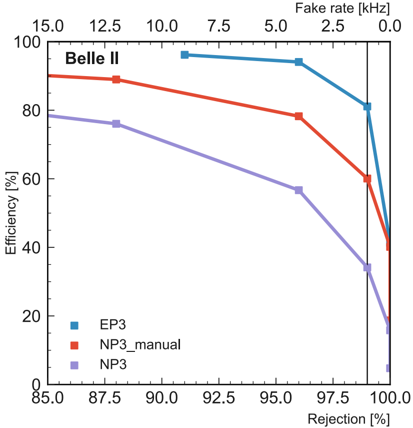
- 10 cluster parameters for each track, so 20 input nodes
- 40 hidden nodes and one classification output node
- Need to choose high neural threshold (here: 0.98) to keep background rate low



Results

- Efficiency acceptable unless vertex displacement too large
- Vertex resolution sufficient for L1 trigger
- Presently not optimized





Summary and Outlook

- Neural Track Trigger fairly optimized and ready to be implemented
- Future training on unbiased data ("f-stream")
- Displaced Vertex Trigger soon to be implemented, but optimization necessary